



EESTI MAAÜLIKOOL

Tartu Tehnikakolledž

**Margus Nurme**

**KAUGJUHITAV ALLVEE VAATLUSSÕIDUK**  
**OBSERVATION CLASS REMOTELY OPERATED**  
**UNDERWATER VEHICLE**

Rakenduskõrghariduse lõputöö

Tehnotroonika õppekava

Juhendaja: dotsent Veli Palge, DrSc

Tartu 2018

## LÜHIKOKKUVÕTE

Eesti Maaülikool Kreutzwaldi 1, Tartu 51014		Rakenduskõrgharidusõppe lõputöö lühikokkuvõte	
Autor: Margus Nurme		Õppekava: Tehnotroonika	
Pealkiri: Kaugjuhitav alvee vaatlussõiduk			
Lehekülgi: 149	Jooniseid: 50	Tabeleid: 1	Lisasid: 6
Osakond: Tartu Tehnikakolledž Uurimisvaldkond: Projekt Juhendaja(d): Veli Palge Kaitsmiskoht ja -aasta: Tartu, 2018			
<p>Hoolimata, et vesi on kõikjal meie ümber, teab inimkond veealusest keskkonnast siiani võrdlemisi vähe. Tänapäevaks on tehnoloogia andnud meile võimaluse seda muuta. Üha rohkem populaarsust on kogumas mehitamata kaugjuhitavate allveesõidukite arendus. Kahjuks pole Eestis antud valdkond siiani suuremat kajastamist leidnud. Käesoleva töö eesmärgiks oli projekteerida oma isiklikuks kasutuseks terviklik allveesõiduki prototüüp ning seeläbi anda põhjalik ülevaade sellise süsteemi mehaanilisest ja elektroonilisest kujundamisest.</p>			
Märksõnad: ROV, hüdrodünaamika, vesi, üleslükkejõud, veealune rõhk			

## ABSTRACT

Estonian University of Life Sciences Kreutzwaldi 1, Tartu 51014		Summary of a Professional Higher Education Thesis	
Author: Margus Nurme		Speciality: Technotronics	
Title: Observation class remotely operated underwater vehicle			
Pages: 149	Figures: 50	Tables: 1	Appendixes: 6
Department: College of Technology Field of research: Project Supervisors: Veli Palge Place and date: Tartu, 2018			
Regardless, that water is found everywhere around us, humankind still knows very little of subsea environment. With todays increasing popularity and rapid development of remotely operated underwater vehicles, technology has given us an opportunity to change that. Unfortunately, in Estonia this field is yet to gain any public attention. The aim of this thesis was to give complete overview of all the mechanical and electrical aspects of designing a fully functional remotely operated underwater vehicle for personal use.			
Keywords: ROV, hydrodynamics, water, buoyancy, subsea pressure			

# SISUKORD

SISSEJUHATUS .....	9
1. TÖÖ EESMÄRK.....	11
2. VALDKONNA ISEÄRASUSED JA SEADME SPETSIFIKATSIOON.....	13
2.1. Nõuded mehaanikale.....	13
2.2. Nõuded elektroonikale .....	14
2.3. Nõuded juhtprogrammile .....	14
3. SEADME MEHAANILINE DISAIN JA KIRJELDUS .....	16
3.1. Korpus.....	16
3.1.1. Geomeetria.....	17
3.1.2. Mootorite paigutus.....	19
3.1.3. Allveesõiduki käitumine vees.....	25
3.1.4. Stabiilsus ja komponentide paigutus .....	26
3.1.5. Tihendus .....	28
3.1.6. Üldine mehaaniline analüüs.....	31
3.1.7. Mehaaniline tugevus ja vastupidavus hüdrostaatilisele rõhule.....	35
3.1.8. Allveesõiduki hüdrodünaamiline analüüs .....	39
3.1.9. Disaini progress .....	40
3.2. Kondensatsioon.....	41
4. SEADME KOMPONENTIDE KIRJELDUS.....	42
4.1. Propulsiivseade .....	42
4.1.1. Mootorid .....	42
4.1.2. Sõukruvid .....	46
4.1.3. Düüs.....	47



4.1.4.	Elektroonilised kiiruse kontrollid .....	48
4.1.5.	Käitursüsteemi test .....	49
4.2.	Läbiviigud .....	53
4.3.	Kaamerakambri aken .....	54
4.4.	Sidevahendid .....	55
4.4.1.	Meediakonverter .....	55
4.4.2.	Kaabel .....	56
4.5.	Kaamera .....	58
4.6.	Juhtpult .....	60
4.7.	Akupakk .....	60
4.8.	Valgustid .....	61
5.	SEADME RIISTVARALINE DISAIN .....	63
5.1.	Kontrollerplaat .....	64
5.1.1.	Elektritoite ahelad .....	65
5.1.2.	Mõõteahelad .....	65
5.1.3.	Andurid .....	66
6.	SEADME TARKVARA .....	68
6.1.	Mikrokontroller .....	68
6.2.	Raspberry Pi .....	68
6.3.	Käitursüsteemi testprogramm .....	70
7.	TULEMUSED JA ANALÜÜS .....	71
7.1.	Disainitud allveesõiduk .....	71
7.2.	Disainitud kontrollerplaat .....	72
7.3.	Tarkvara .....	72
7.4.	Võimalikud edasiarendused .....	72
	Kokkuvõte .....	73
	KASUTATUD KIRJANDUS .....	74

LISA A – Allveesõiduki korpus ja selle detailid .....	77
LISA B – Arvutused .....	99
LISA C – Kontrollerplaadi elektroonikaskeem .....	107
LISA D – Mikrokontrolleri püsivara .....	114
LISA E - pardaarvuti tarkvara .....	127
LISA F - Käitursüsteemi testprogramm .....	145

## TÄHISED JA LÜHENDID

AUV – *Autonomous underwater vehicle*, mehitamata autonoomne veealune sõiduk

Bootloader – väiksemahuline programm, mis käivitub mikrokontrolleri taaskäivitamisel

CAD - *Computer-aided design*, raalprojekteerimine

ESC – *Electronic speed controller*, elektrooniline kiiruse kontrollerr

FOC – *Field oriented control*, vektoriaalne kiirusekontrollimise meetod

Full HD – *Full high definition*, täiskõrglahutus

HIPS – *High impact polystyrene*, löögikindel polüstüreen

I<sup>2</sup>C – *Inter-integrated circuit*, kahesuunaline kahesooneline järjestiksiin

IMU – *Inertial measurement unit*, inertsiaalandur

Kavitatsioon – auru, gaasi või nende segu sisaldavate tühemike (mullide) moodustumine  
voolavas vedelikus

Kv markeering – tähis, mis näitab mootori pöörlemiskiiruse sõltuvust mootori klemmidele  
rakendatud pingest

LDO – *Low-dropout regulator*, madala lävivaruga alalispingeregulaator

LEM – Lõplike elementide meetod

Limonene - naturaalne, tsitruseliste koortest leiduv aine

MATE – *Marine Advanced Technology Education Center*

MJPEG – *Motion Joint Photographic Experts Group*, multimeedia vormingute alamhulk

P2P – *Peer-to-peer*, partnervõrk

PDI – *Program and debug interface*, programmeerimis ja silumis liides

POM – *Polyoxymethylene*, poliüksümetüleen

PWM – *Pulse with modulation*, pulsilaiusmodulatsioon

ROV – *Remotely operated underwater vehicle*, mehitamata kaugjuhitav allveesõiduk

Shore duromeeter – mõõtevahend elastsete materjalide kõvaduse määramiseks

SPI – *Serial peripheral interface*, sünkroonse järjestiksuhtluse liidese standard

USB – *Universal serial bus*, universaalne järjestiksiin

UVC – *USB video class*, USB video seadmete klass

WebRTC - *Web real-time communication*, vabavaralistel rakendusliidestel põhinev reaajaline suhtlusviis

YUY2 – värvide kodeerimise süsteem

$C_d$  – takistuse koefitsient

## SISSEJUHATUS

Mehitamata kaugjuhitav veealune sõiduk (ingl k *remotely operated underwater vehicle*, ROV) on mobiilne seade, mis võimaldab selle operaatoril reaalajas läbi kasutajaliidese vaadelda veealust keskkonda. Sõiduki liikumis- ja manööverdamisvõime tagavad sõidukil paiknevad mootorid ning veealuse pildi edastamiseks on sõidukile paigaldatud kaamera. Lisaks võib allveesõidukil paikneda veel mitmeid andureid ja manipulaatoreid. Üldjuhul pole sellised sõidukid autonoomsed ning kogu andmevahetus (videosignaal, andmed anduritelt ja juhtimine) käib läbi kaabli, mis on ühendatud sõiduki ja veest väljaspool paikneva juhtliidese vahele. Allveesõiduki tööhoidmiseks vajaminev energia tagatakse kas pardal olevatest akumulaatoritest või läbi kaabli. [1]

Kõige sagedamini on allveesõidukid kasutusel süvavee tööstustes ning erinevaid veealuseid uuringuid läbiviivates ettevõtetes. Sellise otstarbega seadmed on enamasti väga suured, rasked ja kallid. Üha enam koguvad populaarsust aga nõ „vaatlusklassi“ veealused allveesõidukid, mis on oma mõõtmetelt tunduvalt väiksemad ja ka kergemad, kui eelpool mainitud tööstusklassi seadmed. Tavaliselt jääb sedalaadi seadmete maksimaalne töösügavus alla 300m. Olgugi, et sellised seadmed oleks üksikisikule või harrastajatele juba käepärasemad, on nende turuhind siiski küllatki kõrge – keskmise sõiduautoga samas hinnaklassis. Seetõttu on praeguseks välismaal välja kujunenud küllaltki suured allveesõidukite ehitajate kommuunid, kuhu kuuluvad mitmesuguste valdkondade inimesed, kes hobikorras töö või õpingute kõrvalt endile allveesõidukeid ehitavad. Tuntumad nendest on [homebuiltrov.com](http://homebuiltrov.com), [rcgroups.com](http://rcgroups.com), [rov.org](http://rov.org) ja [forum.openrov.com](http://forum.openrov.com). Lisaks on allveesõidukite ehitamine mujal maailmas väga populaarne õpilaste ja üliõpilaste seas, kes iga-aastaselt võistlevad enda ehitatud allveesõidukitega MATE võistluste sarjas.

Allveerobotite ehitamise kultuur ja kommuun Eestis praktiliselt puudub, mistõttu on asjakohast informatsiooni küllaltki raske leida. Veebis otsingumootoreid kasutades võib leida infot ainult Tallinna Tehnikaülikoolis loodud autonoomse allveeroboti U-CAT kohta, mis on spetsiaalselt allveearheoloogidele väljatöötatud laevavrakkide uurimiseks [2]. Veel

võib ka teada saada seda, et Eesti mereinstituut on uuringute teostamiseks soetanud SeaBotix ning Muinsuskaitseamet VideoRay firma poolt toodetud allveesõiduki [3].

# 1. TÖÖ EESMÄRK

Käesoleva töö idee sündis tänu Tartu Ülikooli aine „Riistvaraprojekt“ õppejõule, kelle juurde töö autor pöördus ideega teha aine kursusetöök kalastamisel kasutatav allveekaamera, mida tamiili peale kinnitatuna oleks võimalik vette loopida ning seeläbi näha, kus kalad paiknevad. Õppejõu nägemus oli, et selliseks puhuks oleks otstarbekam kasutada mingisugust laadi robotit. Hiljem sama mõtet edasi arendades ning avastades allveesõidukite laadsete masinate olemasolu, tundus taolise seadme disainimine ainuõige valik. Sellest tulenevalt on käesoleva lõputöö eesmärgiks disainida ja võimalusel ka valmis ehitada allveesõiduki kontrollerplaat ja prototüüp, mis realiseeriks tehnoloogiliselt kaasaegseid lahendusi, oleks kompaktne, võrdlemisi taskukohane ning tavakasutajale lihtsasti käsitletav ja lõbus sõiduvahend. Idee kasvades leiti veelgi põhjuseid sellise seadme väljaarendamiseks:

- autori isiklik huvi paremini mõista kalade käitumismaneere ning vees elutsevate organismide elukeskkonda;
- soov saada aimu, mida kujutab endast ühe süsteemi, kui terviku väljatöötamine;
- kontrollida ning ühtlasi ka endale tõestada, et tehnootronika eriala läbinud üliõpilane on võimeline looma mehaaniliselt, elektrooniliselt ja tarkvaraliselt töötava tehnilise terviklahenduse;
- soov käsitleda valdkonda, mis on Eestis küllaltki tundmatu ning kus pole ametlikult praktiliselt midagi loodud;
- saada uusi ja täiendada seniomandatud insenerile vajalikke teadmisi;
- iseehitamine on huvitavam ning odavam, kui allveesõiduk poest ostmise.

Käesolev töö annab põhjaliku tehnilise ülevaate tervikliku allveesõiduki süsteemi mehaanilisest, elektroonilisest ning tarkvaralisest kujundamisest.

Loodetavasti aitab antud valdkonda käsitleva lõputöö koostamine ka Eestis allveesõidukite laadsete seadmete olemasolu rohkem teadvustada. Eestlasedki võiksid korraldada õpilaste vahelisi mõõduvõtmisi allveesõidukite ehitamises ning osaleda ka rahvusvahelisel MATE

võistlusel. Andekamatele üliõpilastele võiks see kujuneda lävepakuks, et töötada tulevikus mõnes antud valdkonnaga tegelevas ettevõttes.



## **2. VALDKONNA ISEÄRASUSED JA SEADME SPETSIFIKATSIOON**

Veekeskkonnas funktsioneeriva seadme koostamine on oluliselt erinev maa peal ja õhus töötavate masinate ehitamisest. Vesi on keeruline keskkond ning selles töötamiseks mõeldud sõiduki loomine esitab ehitajale mitmeid iseäralikke väljakutseid, millest kõige keerulisemad on eelkõige mehaanilist laadi. Toimiva süsteemi kujundamiseks tuleb mõista veekeskkonna iseärasusi. Tuleb arvestada paljude teguritega nagu valguse puudulikkus, halb nähtavus, valguskiirte murdumise mõju optilistele seadmetele ning ka välisjõududega nagu rõhk, vee liikumine, üleslükkejõud ja veetakistus. Vee hea elektriühitavuse tõttu on ka elektriliste komponentide kasutamine riskantne ja raskendatud. Lisaks ei taha vees levida raadiolained, mis raskendab allveesõidukiga side pidamist [1].

Allveesõiduki kujundamisel tuleb kindlasti arvestada ka komponentide massi ning seadme sees paiknevate õhku sisaldavate anumate tühja ruumi suhtega. Kui anumates olev õhk aitab sõidukit pinnal ujutada, siis sisemised komponendid mõjuvad põhja suruva raskusena. Allveesõiduki struktuurist, kujust ja materjalist sõltub vastupidavus veealusele rõhule, võimalik liikumiskiirus, stabiilsus ning manööverdamisvõime.

Korraliku seadme disainimise protsessi loomulikuks eelduseks on paika panna võimalikult detailne tehniline kirjeldus ja nõuded, millele seade vastama peaks.

### **2.1. Nõuded mehaanikale**

Disainitav allveesõiduk peab mehaaniliselt vastama järgmistele nõuetele:

- võimaldama laskuda vähemalt 50 meetri sügavusele veepinnast, st õhuga täidetud komponendid peavad taluma neile 50 meetri sügavusel mõjuvat rõhku;
- normaaltingimustes suuteline liikuma kiirusega vähemalt 1 m/s;

- mõõtmelt võimalikult väike ja kaalult piisavalt kerge, et oleks hõlpsasti kaasaskantav (mahuks kohvrissi ja kaaluks alla 5 kg);
- kujult võimalikult hüdrodünaamiline ja madala kerepinna karedusega, et vähendada liikumisel mõjuvat takistust ja liikumiseks kuluvat energiat;
- vees stabiilne, loodis ning heljuvas olekus;
- elektroonikaseadmeid sisaldavate mahutite sisu peab olema vee sissetungi eest kaitstud;
- vajadusel paranduste tegemiseks kohapeal lihtsalt demonteeritav.

50-meetrine sukeldumissügavus seati eesmärgiks seetõttu, et Eesti tingimustes peaks see võimaldama sukelduda enamikes veekogudes, pakkudes seejuures ka küllaldast reservi.

## **2.2. Nõuded elektroonikale**

Seade peab töötama akutoitelt, mis jääb pingevahemikku 7.4-14.4V ning seejuures võimaldama vähemalt ühe tunni pikkust tööaega. Elektriahelate töökindluse ja ohutuse tagamiseks peab allveesõiduki ja maapealse juhtüksuse vahel olema mitteelektriline andmeedastusviis. Disainitav kontrolleri peab võimaldama:

- kontrollida kuni kuute alalisvoolumootori kiirusekontrollerit;
- mõõta aku pinget ning mootorite voolutarvet;
- suhtlust erinevate täiturite, andurite ning Raspberry Pi ja teiste samalaadsete miniarvutitega;
- stabiilset toitepinget kõigile seadme elektroonilistele komponentidele nagu pardakaamera, LED valgustid, Raspberry Pi, meediakonverter ja andurid;
- USB kaudu programmeerimist, isegi kui põhitoiteallikas on ühendatud.

## **2.3. Nõuded juhtprogrammile**

Välja töötada tarkvara beetaversioon, mis esialgu täidab ainult kõige elementaarsemaid tingimusi allveesõiduki kasutamiseks – andmevahetus kontrolleri ja peamise juhtüksuse Raspberry Pi vahel, mootorite kontrollimine juhtpuldiga ning videopildi edastus reaajas.

Kogu tarkvara peab olema allveesõiduki keskne ja juhtliides veebilehitseja põhine, et puuduks vajadus kasutajapoolseks tarkvara paigaldamiseks.

### 3. SEADME MEHAANILINE DISAIN JA KIRJELDUS

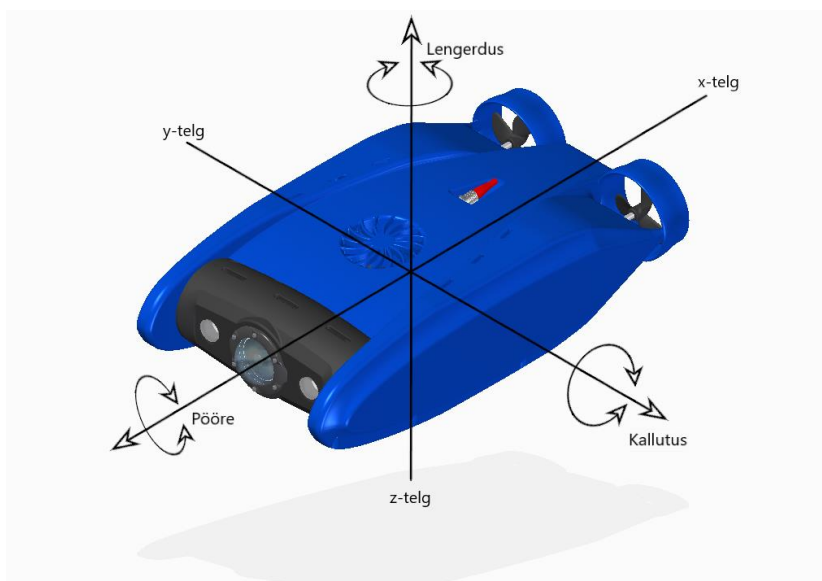
Käesolevas peatükis kirjeldatakse allveesõidukite ehituse peamisi aspekte ja antakse põhjalik ülevaade projekteeritud prototüübi mehaanilistest disainist ning omadustest. Kõik allveesõiduki joonised on toodud töö lisas A. Mehaanilise disainiga seotud arvutused asuvad töö lisas B.

#### 3.1. Korpuse

Allveesõiduki prototüübi väliskorpuse seinad on 4mm paksusega ning 3D prinditakse kõrgendatud löögitaluvusega polüstüreen lõngast (Formfutura EasyFil HIPS), kasutades sulatatud materjali sadestumise tehnoloogiat (ingl k *fused deposition modelling*, FDM). Materjali valikul sai määravaks, et polüstüreen on: 1) võrdlemisi odav, 2) tugev, 3) löögikindel, 4) lihtsasti töödeldav ning 5) 3D printimisel laialdaselt kasutusel. Lisaks on polüstüreen kerge - tihedus ligilähedane vee tihedusele (mage vesi temperatuuril 4-15°C tihedus: 944-1000 kg/m<sup>3</sup>, kasutatud HIPS lõnga tihedus: 1004 kg/m<sup>3</sup>) [1]. 3D printimisel jäävad materjali sisse pisikesed õhumullid, mis teevad korpuse veest kergemaks, aidates kaasa korpuse ujutavuse saavutamisele. Sellest tulenevalt vajavad detailid veekindluse saavutamiseks ka järeltöötlust - efektiivseks meetodiks on pindade katmine epoksüvaiguga. Lisaks on polüstüreeni võimalik siluda, kas limonene või atsetooni aurudega. Korpuse on mitmest kohast avatud ja veega täituv ning kogu elektroonika on korpuse sisse vettpidavatesse mahutitesse ära jaotatud. Detailide 100% täitehulga juures on 3D prinditud väliskorpuse tühimass õhus 1.130 kg. Korpuse on modelleeritud CAD keskkonnas Solid Edge ST10 ning valmistatakse 3D printeriga Creality CR-10 4S.







### 3.1.1. Geomeetria

Vees liikuva keha pikkuse ja laiuse suhe avaldab otsest mõju keha hüdrodünaamikale – mida pikem ja kitsam on keha, seda väiksem liikumistakistus kehale liikumiskiiruse kasvades mõjub. Liiga pika keha puhul kaotatakse aga stabiilsuses. Lühemale kehale mõjub liikumiskiiruse kasvades suurem liikumistakistus, aga keha on seevastu palju stabiilsem ja parema manööverdamisvõimega kõigis kolmes telje suunas (joonis 3.1). [1]



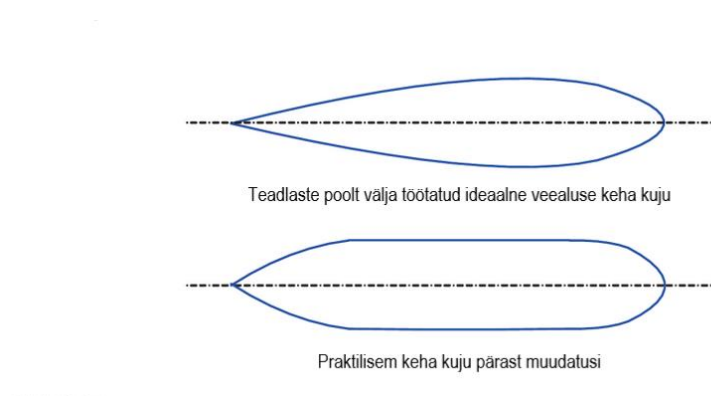
**Joonis 3.1.** Pöörde-, kallutus- ja lengerdusnurgad allveesõiduki juhtimisel.

Allveelaevade disainereid on välja selgitanud, et kõige optimaalsema manööverdamisvõime ja stabiilsuse tagamiseks on parim pikkuse ja laiuse suhe 7:1. Teisalt sõltub vees liikuvale kehale mõjuv takistus keha kujust. Ühe parima tulemuse annab teadlaste poolt nõ „ideaalseks“ hüdrodünaamiliseks kujuks peetav veetilga kujuline ristlõike profiil, mille takistuse koefitsientiks ( $C_d$ ) on 0.04 (joonis 3.2). [1]

Keha kuju		$C_x$
Kera		0.47
Poolkera		0.42
Koonus		0.50
Kuubik		1.05
Pööratud kuubik		0.80
Pikk silinder		0.82
Lühike silinder		1.15
Voolujooneline keha		0.04
Poolitatud voolujooneline keha		0.09

**Joonis 3.2.** Üldlevinud keha kujude takistuse koefitsientid vee all.

Praktikas pole see profiil kuigi kuluefektiivne, mistõttu rakendatakse rohkem selle järgi kohandatud alternatiive (joonis 3.3). [1]



**Joonis 3.3.** Ideaalne veealune keha kuju ja selle kuluefektiivsem modifikatsioon.

Kolmandaks geomeetriliselt kriitiliseks teguriks allveesõiduki disainimisel on keha sümmeetria. Kui mõni osa korpusest on profiiliga, mis põhjustab liikumisel antud kohale märkimisväärselt suuremat takistust, mõjub sinna ka täiendav jõud, mis just liikumiskiiruse kasvades raskendab sirgjoonelist liikumist.

Kuna allveesõiduk on eelkõige mõeldud veealuse keskkonna vaatlemiseks, mitte kiireks ringi sõitmiseks, on üldjuhul nende pikkuse ja laiuse suhe küllaltki väike. Disainitava sõiduki optimaalne pikkuse ja laiuse suhe ning mootorite paigutus sõltub eelkõige seadme

eesmärgist – kui kiiresti on tarvis liikuda, kui kitsastes oludes ning, mis telgesid pidi soovitakse manööverdada.

Korpuse suuruse määramisel lähtuti põhimõttest, et korpus ei tohi põhjuseks olla suurem kui kõigi komponentide mahutamiseks tarvis - mida suurem on vedelikus liikuva keha pind, seda suurem hõõrdetakistus talle mõjub [4]. Seejuures peeti oluliseks saavutada ka piisav voolujoonelisus, mis tagaks sujuva vee liikumise sõukruvideni ja võimaldaks teha liikumisel ka äkilisemaid kiirendusi ning suunamuutusi. Võimalikult mitmekülgse videopildi tagamiseks asub allveesõiduki vööris 136 kraadi ulatuses pöörlev korpus kaamera ja LED valgustite kinnitamiseks. Allveesõiduki prototüübi korpus on 40 cm pikk, 22 cm lai ja 10.4 cm kõrge. See annab pikkuse ja laiuse suhte  $\sim 2:1$ , mis tagab hea stabiilsuse ümberpöörumise vastu. Liikumisel mõjuva takistuse vähendamiseks on rakendatud sümmeetrilist ja veetilga kujule võimalikult sarnast profiili. Täiendavaks kaitseks teiste objektidega kokkupõrke eest on sõiduki külgi laiendatud. Korpuse kuju on inspireeritud SAAB Sabertooth AUV allveeroboti konseptsioonist (joonis 3.4).

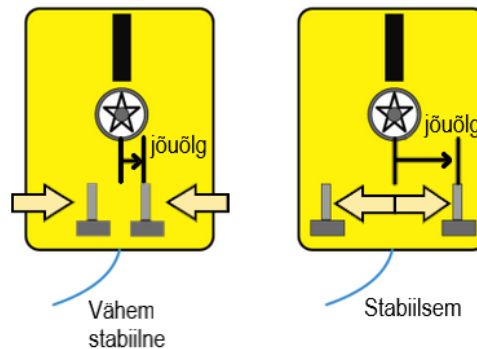


**Joonis 3.4.** Allveesõiduki prototüübi disain.

### **3.1.2. Mootorite paigutus**

Lisaks pikkuse ja laiuse suhtele ning keha kujule, mõjutab allveesõiduki stabiilsust liikumisel ka mootorite asetuse. Mida kaugemal asetsevad üksteisest ja sõiduki

raskuskeskmest horisontaalteljel paiknevad mootorid, seda stabiilsem sõiduk liikumisel on. Seeläbi suureneb allveesõiduki raskuskeskme ning mootori vaheline jõuõlg, mis omakorda parendab manööverdamisvõimet piki lengerdusnurka (joonis 3.5). [1, 4]



**Joonis 3.5.** Allveesõiduki stabiilsus sõltuvalt mootorite asetusest [1].

Ühtlasi, mida kaugemal on mootorid allveesõiduki pöölemiskeskme, seda kiiremini sõiduk pöörab. Ka vertikaalsihis paiknevate mootorite asetus, mida kasutatakse sõiduki töösügavuse muutmiseks ja mõnel puhul ka pööramiseks, mõjutab stabiilsust. Näiteks sukeldudes mõjub vertikaalse mootori poolt tekitatav jõud lisaraskusena kõrgel sõiduki kohal, mis kergitab sõiduki raskuskeset ning destabiliseerib seda pöörde- ja kallutusnurkapidi [1]. Juhul kui ei taodelda võimalust liikuda pöörde- või kallutusnurka pidi, peaksid vertikaalsed mootorid paiknema allveesõiduki raskuskeskmega kohakuti. Mootorite arv ning nende asetus määrab ka allveesõiduki manööverdamisvõime ning liikumise efektiivsuse.

Mootorite paigutamiseks on mitmeid üldlevinud 3, 4 või 5 mootoriga asetusi.

Asetus 3 mootoriga:

- kaks paralleelset horisontaalsuunalist mootorit horisontaalsihis liikumiseks ja üks vertikaalsuunaline mootor vertikaalsihis liikumiseks (joonis 3.6).





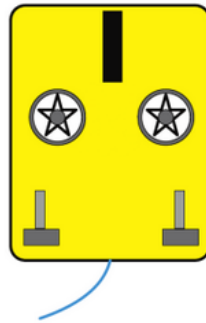
**Joonis 3.6.** Kolme mootoriga allveesõiduki pealtvaade [1].

Antud asetus võimaldab liikuda:

- otse, edasi ja tagasi – mõlemad horisontaalsed mootorid ühtlasel kiirusel;
- edasi või tagasi, samal ajal pöörates paremale ja vasakule – kumbki horisontaalsetest mootoritest töötab üksi;
- keerates mööda lengerdusnurka paremale või vasakule – mõlemad mootorid töötavad ühtlasel kiirusel kuid vastupidises suunas;
- üles ja alla – vertikaalse mootori töötamisel;
- kallutada – juhul kui vertikaalne mootor asub pikisuunaliselt (x-teljel) raskuskeskmest nihutatult.

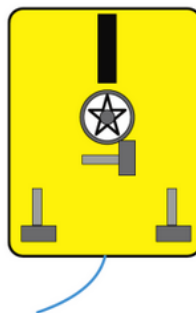
Asetused 4 mootoriga:

- Kaks paralleelset horisontaalsuunalist mootorit horisontaalsihis liikumiseks ja kaks vertikaalsuunalist mootorit vertikaalsihis liikumiseks (joonis 3.7). Antud paigutuse juures tagatakse kahe vertikaalsuunalise mootori ühtlasel tööil lisajõud üles-alla liikumisel või kallutamisel. Lisaks võimaldab kummagi vertikaalse mootori eraldi jooksumine allveesõidukit pöörata.



**Joonis 3.7.** Nelja mootoriga allveesõiduki pealtvaade [1].

- Kaks paralleelset ja üks risti paiknev horisontaalsuunaline mootor horisontaalsihis liikumiseks ning üks vertikaalsuunaline mootor vertikaalsihis liikumiseks (joonis 3.8).

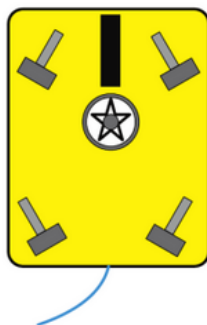


**Joonis 3.8.** Nelja mootoriga allveesõiduki pealtvaade [1].

Antud paigutus võimaldab lisaks eelpool mainitutele ka külg-liikumist, kuid ei võimalda pöörmist.

Asetus 5 mootoriga:

- Neli  $45^\circ$  nurga all paiknevat horisontaalsuunalist mootorit horisontaalsihis liikumiseks ja üks vertikaalsuunaline mootor vertikaalsihis liikumiseks. Rohkem tuntud kui vektoriaalne mootorite asetuse (joonis 3.9).

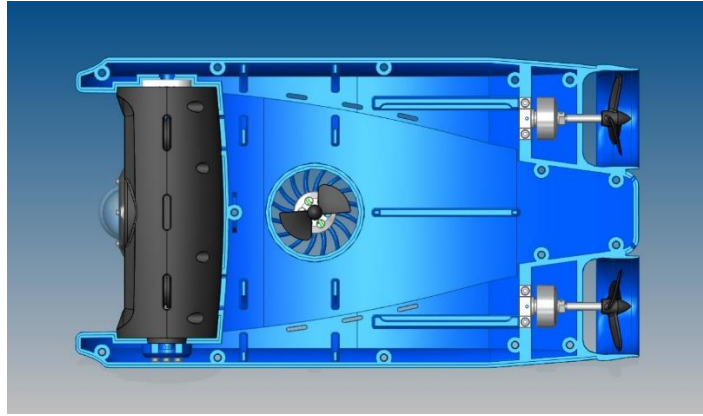


**Joonis 3.9.** Vektoriaalse mootorite asetusega allveesõiduki pealtvaade [1].

Antud paigutus võimaldab liikuda horisontaalselt mis tahes suunas ning seda ka ühe horisontaalse mootori rikke korral.

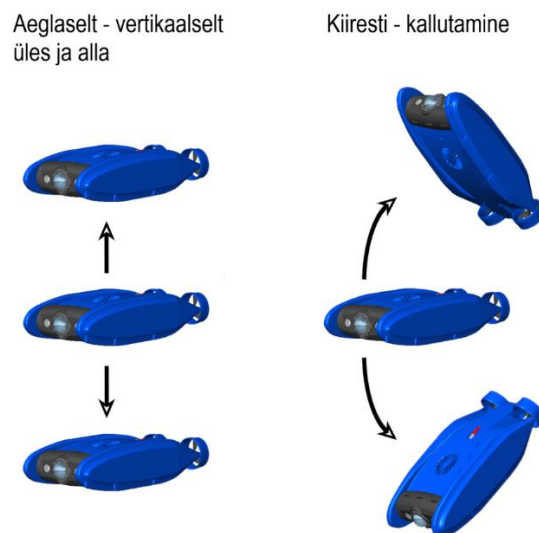
Väljatoodud 3 ja 4 mootoriga paigutused nõuavad vähem mootoreid, aga eeldavad kahe-suunalist kontrolli mootorite üle – st mootoreid peab saama jooksutada nii päri- kui ka vastupäeva. See nõuab mootorite kontrollimiseks keerukamat ja kallimat elektroonilist lahendust ning seegi ei pruugi tagada nii mitmekülgset kontrolli kui vektoriaalne asetuse. Vektoriaalne asetuse pakub mitmekülgsemaid võimalusi manööverdamiseks ja ei nõua tingimata kahe-suunalist kontrolli mootorite üle, kuid selle lahenduse peamiseks puuduseks on mootorite  $45^\circ$  nihe sirgjooneliste liikumissuundade suhtes, mis põhjustab ligi 30% kadu tõukejõu maksimumist [4]. Lisaks nõuab vektoriaalne asetuse ka tunduvalt keerulisemat juhtprogrammi.

Mootorite paigutamisel kaaluti ja disainiti nii 3 kui ka 4 mootoriga lahendusi. Esialgse disaini lihtsustamiseks otsustati siiski 3 mootoriga süsteemi kasuks – 1 vertikaalselt ja 2 horisontaalselt paiknevat mootorit (joonis 3.10).



**Joonis 3.10.** Mootorite paigutus prototüübi korpusel.

Manööverdamisvõime parandamiseks asetsevad horisontaalteljelised tüürpoordi ja pakpoordi mootorid üksteisest võimalikult lahku paigutatult. Vertikaalne mootor asub keskel, kuid sõiduki raskuskeskmest eesotsa suunas nihutatult. Selline asetus võimaldab liikuda vertikaalsihis nii üles-alla kui ka y-telge pidi kallutada – vertikaalset mootorit aeglaselt jooksutades liigub sõiduk üles-alla, kiirelt jooksutades viiakse sõiduk tasakaalust välja võimaldades seda kallutada (joonis 3.11).



**Joonis 3.11.** Keskpunktist nihutatud vertikaalse mootoriga allveesõiduki liikumine vertikaalsihis.

### 3.1.3. Allveesõiduki käitumine vees

Vedeliku sisse asetatud kehale mõjub vedeliku poolt tekitatud jõud, mis seda keha üles lükkab – üleslükkejõud. See jõud mõjub raskusjõule vastupidises suunas ning tuleneb sellest, et vesi proovib täita ruumi, mille keha vees hõivab [1, 4]. Sellest tulenevalt on kolm erinevat võimalust, kuidas keha vedelikus käitub [5]:

1. Täielikult vees paiknev keha tõuseb pinnale ja ujub, kui üleslükkejõud on suurem kui kehale mõjuv raskusjõud. Veepinnal jõud võrdsustavad, sest osa kehast kerkib veest välja.
2. Keha heljub vees, kui üleslükke- ja raskusjõud tasakaalustavad üksteist ehk üleslükkejõud on võrdne kehale mõjuva raskusjõuga.
3. Keha upub ehk vajub põhja, kui üleslükkejõud on väiksem kui kehale mõjuv raskusjõud.

Archimedese printsiip ütleb, et vedeliku sisse asetatud kehale mõjuv üleslükkejõu suurus on võrdne keha poolt kõrvale tõrjutud vedeliku hulga mõjuva raskusjõuga ehk selle vedeliku hulga kaaluga [1].

Allveesõidukite puhul on soovituslik, et temale vees mõjuv üleslükkejõud oleks esialgu kuni 4% suurem, kui talle mõjuv raskusjõud, ehk oleks, mis on natukene heljuvuse piirist ujuvuse kasuks. Seejärel lisatakse vähesel määral raskuseid, et trimmida raskuskeskme ja üleslükkejõu keskme asukohta üksteise suhtes. Lõpuks peaks üleslükkejõud olema 1-2% raskusjõust suurem [4]. Heljuvas olekus kulub vees liikudes keskmiselt kõige vähem energiat, sest mootorite abil ei ole tarvis pidevalt soovitud töösügavust hoida. Kui kumbki nimetatud jõududest teist märkimisväärselt ületab, peab keha sügavust mootorite abil pidevalt korrigeerima, et uppuvat keha kergitada, või pinnale tõusvat alla suruda. Vähene seadistus ujuvuse kasuks on eelistatud, sest juhul kui kaotatakse füüsiline ühendus sõidukiga, ei jää sõiduk vee alla heljuma ega vaju põhja, vaid kerkib ise aeglaselt pinnale. Teiseks põhjuseks on veekogu põhjalähedane liikumine, kus vertikaalselt üles liikumine kergitab põhjas olevat setet ja halvendab nähtavust.

Vedeliku sees oleva keha käitumist saab hinnata mitmel meetodil [4]:

- võrreldes keha kaalu keha poolt kõrvale tõrjutud vedeliku kaaluga;
- võrreldes keha keskmist tihedust vedeliku tihedusega.

Sellegipoolest sõltub keha hõljumine vedelikus eelkõige keha keskmisest tihedusest. Kui keha keskmine tihedus on väiksem vedeliku tihedusest, siis jääb keha vedeliku pinnale ujuma. Kui keha keskmine tihedus on suurem vedeliku tihedusest, siis vajub keha põhja. Võrdsete tiheduste korral jääb keha vedelikus heljuma. Juhul, kui keha on valmistatud vedeliku tihedusest suurema tihedusega materjalist, aga keha sisemuses asub piisavalt suur õhuga täidetud tühimik, siis selle keha keskmine tihedus võib olla väiksem kui vedeliku tihedus [5]. Tavaliselt kasutatakse allveesõiduki keskmise tiheduse langetamiseks kõiksuguseid ujukeid, mis on võrdlemisi suure ruumala juures väga väikese tihedusega. Üldiselt on nendeks, kas gaasiga täidetud mahutid või spetsiaalsest sünteetilisest vahust vormitud osad.

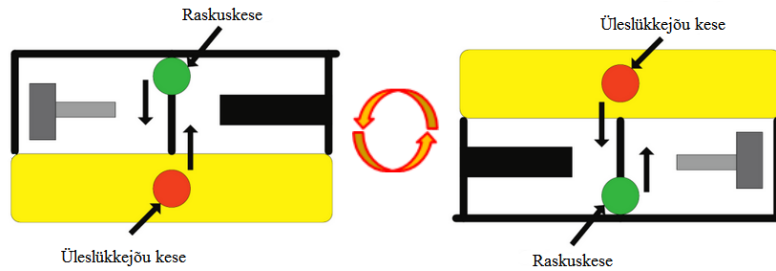
#### **3.1.4. Stabiilsus ja komponentide paigutus**

Selleks, et vältida vees allveesõiduki ümber rullumist või kaldumist mis tahes küljele, tuleb kõik raskused ja ujukid hoolikalt paigutada. Komponentide paigutus sõiduki korpuses mõjutab raskuskeskme ja üleslükkejõu keskme asukohta ning seeläbi ka stabiilsust.

Allveesõiduki raskuskese on punkt, kuhu on koondunud kogu kehale mõjuv raskusjõud [1].

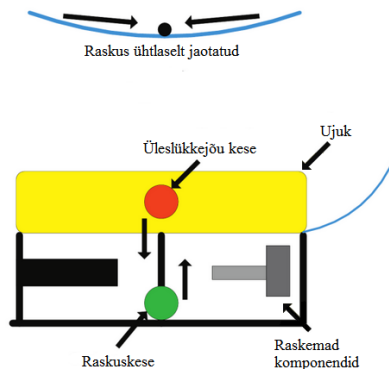
Allveesõiduki üleslükkejõu kese on punkt, kuhu on koondunud kogu kehale mõjuv üleslükkejõud [1].

Ideaalis tuleks paigutada kõik raskused sõiduki põhjale võimalikult lähedale ning ujukina toimivad kehad raskuste suhtes võimalikult kõrgele - seejuures on äärmiselt tähtis, et asetus oleks sümmeetriline. Nendel tingimustel püsib raskuskese madalal, üleslükkejõu kese kõrgel ja mõlemad jõud vertikaalselt samal joonel ning üksteisest võimalikult kaugel. Kuna vedelikus paiknev keha pöörab ennast alati nii, et üleslükkejõu kese oleks raskuskeskme kohal, tuleb olla kindel, et raskuskese paikneb üleslükkejõu keskmest all pool (joonis 3.12) [1, 4].



**Joonis 3.12.** Allveesõiduki üleslükkejõu kese pöörduv alati üles, raskuskeskmest kõrgemale [1].

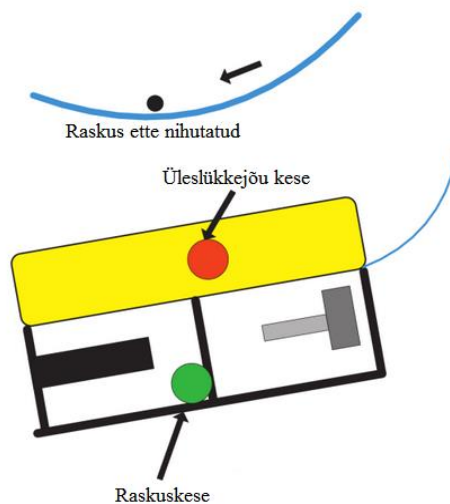
Selle tagamiseks peavad ujuvust tekitavad osad (veest väiksema tihedusega) paiknema raskuste kohal (kõrgemal) (joonis 3.13). Selle vastu eksides rullub sõiduk ümber ning jääb vette tagurpidi. [1]



**Joonis 3.13.** Veis stabiilse allveesõiduki raskuskeskme ja üleslükkejõu keskme asukohad [1].

Mida suurem on raskuskeskme ja üleslükkejõu keskme vahe, seda raskem on keha ümber ajada.

Asümmeetrilise või läbimõtlemta raskustjaotusega ja ujukite paigutusega sõiduk hoiab vees viltu, kuna kumbki keskmest nihkub teisest mingi külje suunas (joonis 3.14).



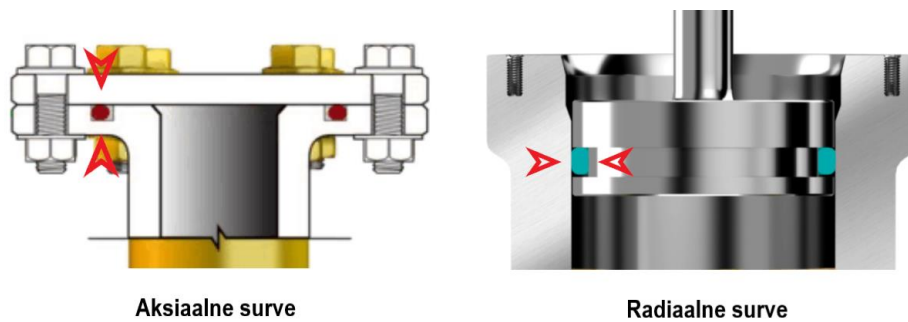
**Joonis 3.14.** Allveesõiduki raskuse ja üleslükkejõu kese nihkumine asümmeetrilise komponentide paigutuse korral [1].

Üldjuhul pole täiesti stabiilse allveesõiduki saavutamine võimalik, kasutades ainult süsteemi toimimiseks vajalike komponente. Kehale mõjuva üleslükke- ja raskusjõu suhe kipub alati ühe või teise kasuks olema. Samuti ei saa olla kõik komponendid ja nende asetus täiesti sümmeetriline. Eelpool mainitud hälvete kompenseerimiseks tuleb erinevatesse kohtadesse lisada täiendavaid ujukeid või raskuseid, mis sõiduki funktsionaalsuse kohapealt suurt tähtsust ei oma. [1]

### 3.1.5. Tihendus

Vältimaks vee ligipääsu elektroonikale 50 meetri sügavusel 490 kPa suuruse rõhu juures, kasutatakse mahutite tihendamiseks o-rõngastihendeid. See on lihtne, odav ja efektiivne moodus töökindla tihenduse tagamiseks. Kasutatud on kahte staatilist tüüpi ühendust - aksiaalse ehk teljesihilise ja radiaalsuunalise survega. Aksiaalse liite korral pigistatakse tihendi ristlõike ülemist ja alumist pinda. Radiaalse liite korral pigistatakse tihendit tema sise- ja välisraadiuse suunast (joonis 3.15).





**Joonis 3.15.** O-rõngastihendi aksiaalne ja radiaalne surve.

O-rõngastihendite kasutamisel on kriitiliselt oluline korrektne tihendi soone disain ja tihendiga kokkupuutuvate pindade kvaliteet. Tihendi soone sügavus ja laius peab olema liite tüübile ning tihendi läbimõõdule ja kujule vastav. Tihendite valikul ja soonte disainil on järgitud Parker o-rõngasnööri käsiraamatu tööstusliku klassifikatsiooni nõudeid, mille kohaselt spetsifikatsioonile vastava disaini survetaluvus on kuni 10.35 Mpa. [6]

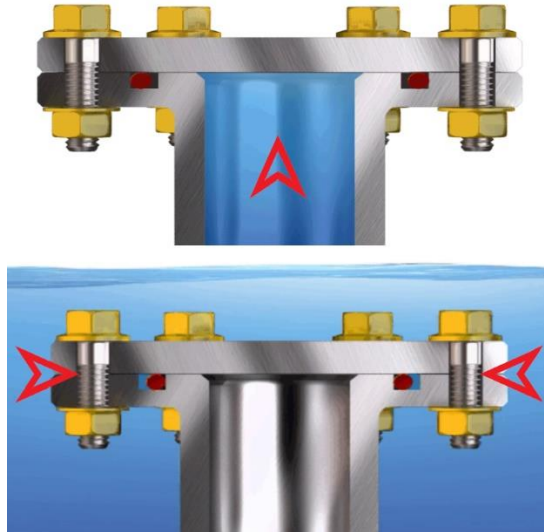
Staatilisel ühendusel on tihendiga kontaktis olevate pindade soovituslik pinnakaredus 1.6  $\mu\text{m}$  gaaside ja kuni 3.2  $\mu\text{m}$  vedelike puhul [6]. Prototüübi tihendite sooned on 3D prinditud, mistõttu tuleb neid soovitusliku pinnakareduse saavutamiseks järeltöödelda (lihvida või kemikaalidega siluda). Kehva pinnakvaliteedi kompenseerimiseks on kasutatud võimalikult suure läbimõõduga elastset tüüpi tihendeid, mis on stabiilsemad ja soone kvaliteedi suhtes vähem nõudlikud [6]. Kõik kasutatud tihendid on valmistatud nitrilkummist kõvadusega 70 Shore durometri A skaalal ning on ristlõike läbimõõduga 1.78 kuni 3.53 mm (tabel 1).

**Tabel 3.1.** Allveesõiduki o-rõngastihendid ja nende soone mõõtmed

Asukoht	Siseläbim õõt, mm	Ristlõi ke läbim õõt, mm	Soone sügav us, mm	Soo ne laiu s, mm	Mater jal	Kõvad us, A	Tüüp	Soone sise. Ja välisnur kade raadius, mm
Kaamera- kambri esine	41	1.78	1.32	2.24	NBR	70	aksiaa line	0.25
Kaamera- kambri tagune	-	1.78	1.32	2.24	NBR	70	aksiaa line	0.25
LED valgusti kaas	15.6	1.78	1.3	2.4	NBR	70	radiaa line	0.25
Akukambri kaas	40.94	2.62	3.56	2.05	NBR	70	radiaa line	0.25
Elektroonika kamber	-	3.53	4.1	2.7	NBR	70	aksiaa line	0.5

Nitriilkumm on üldlevinud ja odav o-rõngastihendi materjal, mis on vastupidav ning sobilik veega kokkupuutuvate pindade tihendamiseks [6]. Vältimaks tihendite vigastamist, on kõik teravad soonte nurgad ümardatud. Elektroonika korpuse ja kaamerakambri taguse kujust tingituna on nende tihendid valmistatud o-rõngasnöörist.

Tihendi asukoht soones sõltub suunast, millest tihendile rõhku avaldatakse - tihend peab alati asetsema sellele soone küljele lähemal, kus talle avaldub madalam rõhk (joonis 3.16) [6].



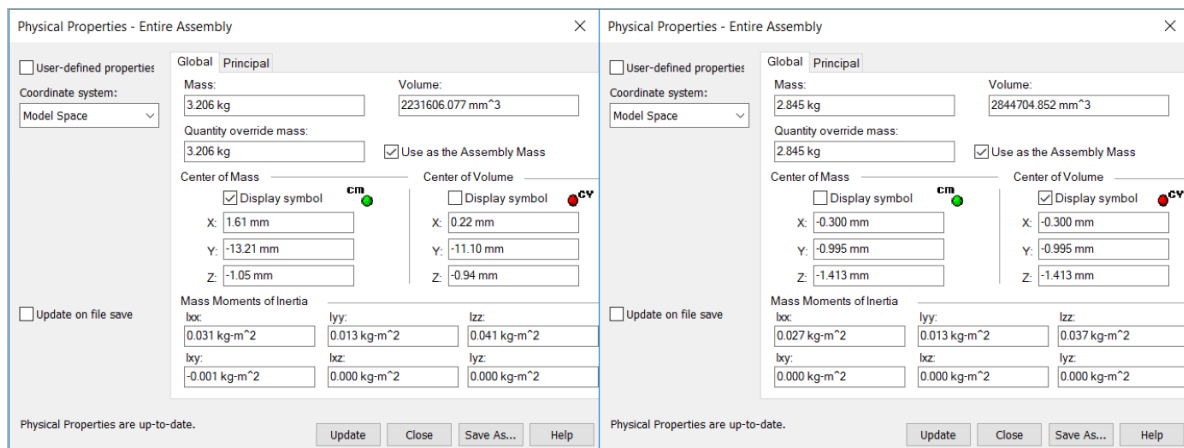
**Joonis 3.16.** O-rõngastihendi asukoht soones, sõltuvalt rõhu avaldumise kohast.

Seega määrab seest poolt avalduva rõhu puhul tihendi läbimõõdu soone välimine diameeter ning väljast poolt avalduva rõhu puhul soone sisemine diameeter. Selle järgimine võimaldab kasutada ka soovitatust kitsamat või laiemat soont ning praktiliselt välistab tihendi liikumise soones. [6]

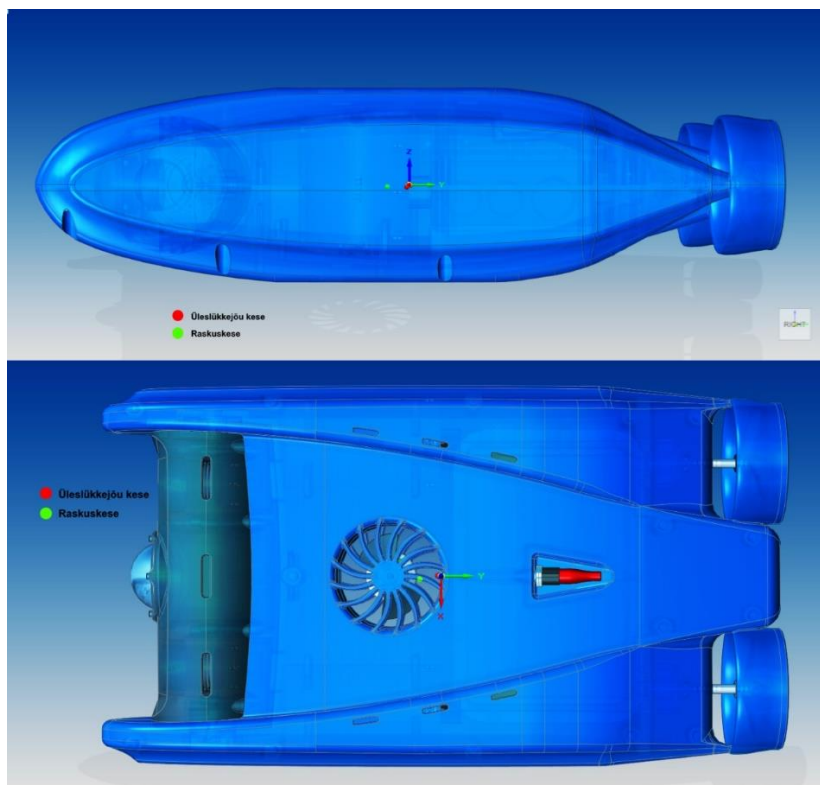
Lõpliku disaini valmimisel freesitakse tihendatud mahutid alumiiniumist. Aksiaalsete ühenduste survestamiseks ja radiaalsete ühenduste fikseerimiseks kasutatakse erineva pikkusega M2 ja M3 keermega sisekuuskantpolte, mis kinnituvad plastiku sisse kuumsurutud tõmbemutritesse.

### 3.1.6. Üldine mehaaniline analüüs

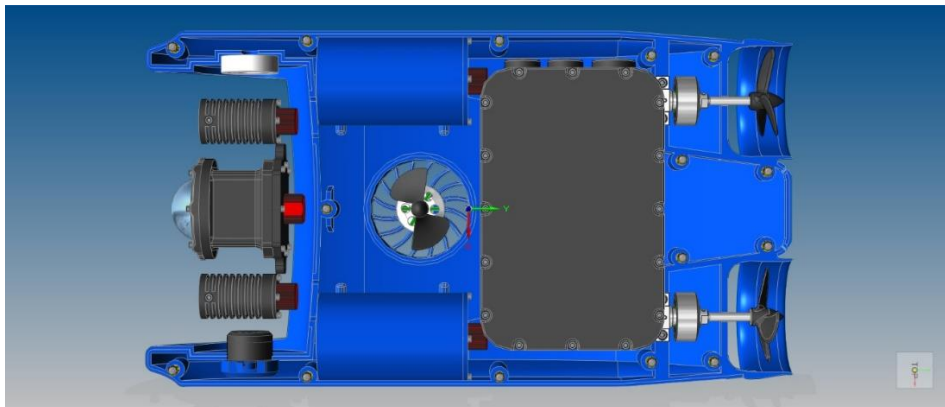
Massiga 3.206 kg ja täielikult vees paikneva allveesõiduki poolt kõrvale tõrjutud vee massiks on 2.845 kg. Mõjuvad raskus- ja üleslükkejõud on vastavalt 31.440 N ning 27.900 N. Allveesõiduki keskmiseks tiheduseks on  $1163.983 \text{ kg/m}^3$ , mistõttu disainitud sõiduk on vees uppuv ning heljuvale ligilähedase oleku saavutamiseks tuleb lisada täiendavaid ujukeid. Kuigi kõik korpuses paiknevad komponendid paigutati võimalikult sümmeetriliselt korpuse põhja, et tagada võimalikult ühtlane raskusjaotus ja madal raskuskese, olid raskuskese ja üleslükkejõu kese teineteise suhtes nihkes ning vertikaalsihis praktiliselt samal kõrgusel (joonis 3.17, joonis 3.18 ja joonis 3.19).



**Joonis 3.17.** Allvesõiduki mass (vasakul) ja tema poolt kõrvale tõrjutud vee mass (paremal) ning nende parameetrid enne Divinycell H vahu ja trimmimis raskuste lisamist.

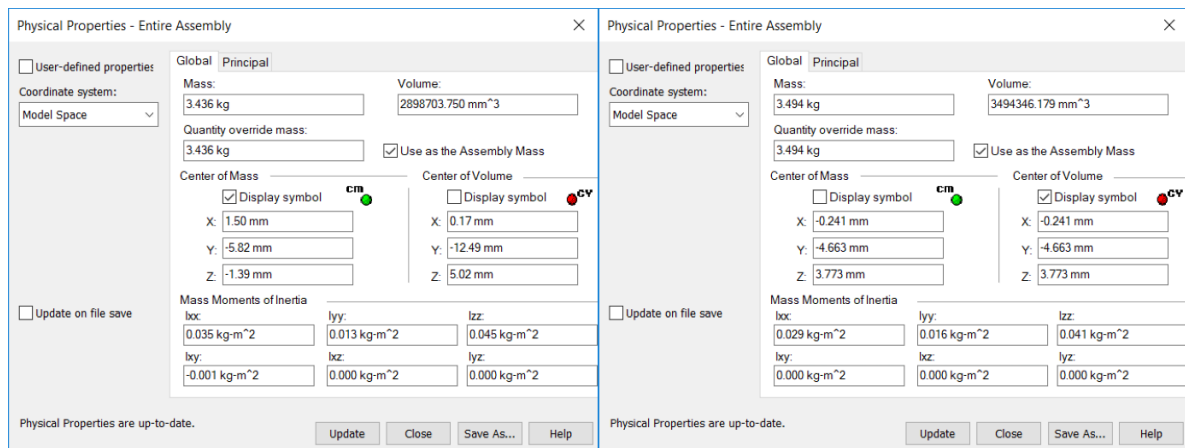


**Joonis 3.18.** Allvesõiduki raskuskeskme ja üleslükkejõu keskme asukoht enne Divinycell H vahu ning trimmimis raskuste lisamist.

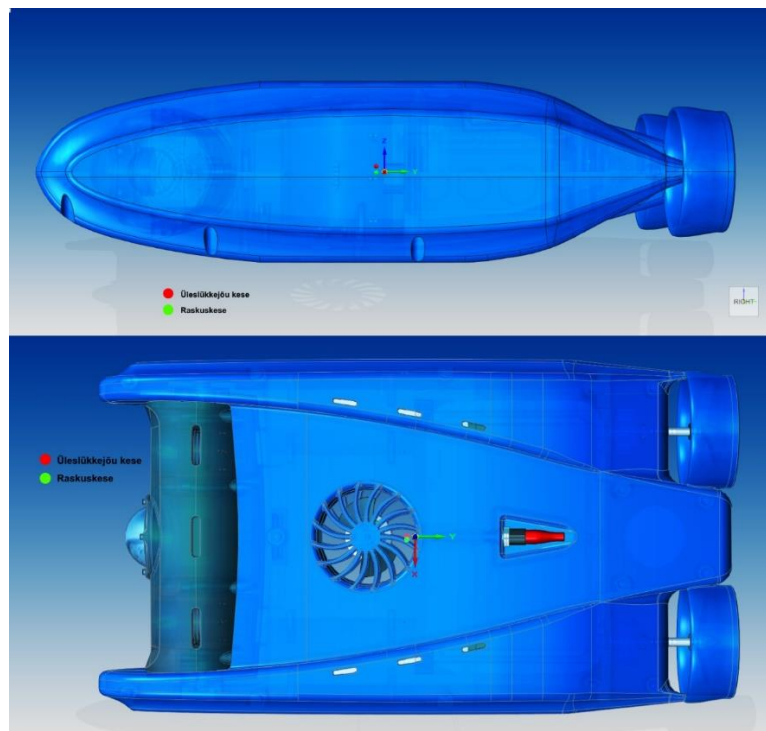


**Joonis 3.19.** Komponentite paigutus allveesõidukis.

Üleslükkejõu tõstmiseks ning tasakaalu korrigeerimiseks kasutati rõhukindlat vahtu Divinycell H tihedusega  $100 \text{ kg/m}^3$ , mida esialgsete arvutuste kohaselt tuli lisada  $543 \text{ cm}^3$ . Antud mahu juures tõusis allveesõidukile mõjuv üleslükkejõud ligikaudu 4% üle vees hõljumiseks vajaliku piiri. Allveesõiduki täiendavaks trimmimiseks tuli ahtrisse lisada 165 grammi raskust, mida oli natukene rohkem kui eeldatud. Selle kompenseerimiseks lisati veel  $107 \text{ cm}^3$  Divinycell H vahtu, pärast mida ületab vee poolt tekitatav üleslükkejõud allveesõidukile mõjuva raskusjõu ligikaudu 1.7% võrra. Ujuki lisamisel arvestati, et ka see suurendab sõidukile mõjuvat raskusjõudu. Tulemusena tõusis sõiduki mass  $0.230 \text{ kg}$  ja tema poolt kõrvale tõrjutud vee mass  $0.649 \text{ kg}$  ning mõjuvad raskus- ja üleslükkejõud vastavalt  $2.256 \text{ N}$  ning  $6.365 \text{ N}$  võrra. Ühtlasi nihkus raskuskese üleslükkejõu keskmest alla poole ja vertikaalsihis praktiliselt ühele joonele (joonis 3.20 ja joonis 3.21).



**Joonis 3.20.** Allvesõiduki mass (vasakul) ja tema poolt kõrvale tõrjutud vee mass (paremal) ning nende parameetrid pärast Divinycell H vahu ja trimmimis raskuste lisamist.



**Joonis 3.21.** Allvesõiduki raskuskeskme ja üleslükkejõu keskme asukoht pärast Divinycell H vahu ning trimmimis raskuste lisamist.

Kindlasti väheneb tegelik trimmimiseks lisatav raskuste kogumass vähemalt poole võrra, sest sõiduki koostamisel lisanduvad süsteemile mitmed väiksemad komponendid nagu elektroonilised mootorite kiiruse kontrollid, silikageeli pakid, tihendid ja juhtmed. Valdav

osa nendest komponentidest läheb allveesõiduki ahtrisse ja neid pole antud modellatsiooni ning arvutuste käigus arvesse võetud.

Kujundatud süsteemi kompleksusest tingituna on eelnev analüüs ja jõudude keskmete korrigeerimine läbi viidud Solid Edge keskkonnas. Raskuskeskme arvutamiseks defineeriti kõigi komponentide materjalide tihedused, paigutati komponendid korpusesse ning arvutati allveesõiduki raskuskese. Komponentid millele tihedust polnud võimalik määrata, kaaluti ära. Üleslükkejõu keskme leidmiseks määrati kõigi komponentide materjalidele vee tihedus ning arvutati allveesõiduki poolt kõrvale tõrjutud vee mass ja selle raskuskese. Sellel juhul on allveesõiduki üleslükkejõu kese arvutatud raskuskeskmega võrdne, sest vedelikus paiknev keha väljutab vedelikust sama geomeetria võrdväärse mahu ja kujuga vedeliku massi, mille raskuskeskme asukoht on võrdne kehale mõjuva üleslükkejõu keskme asukohaga [4]. Seejuures tuli kõik seest õõnsad mahutid täita materjaliga, et programm arvestaks kehade täisruumala. Keskmise tiheduse leidmisel oli oluline arvestada õõnsates mahutites paikneva õhuga. Õhu hulga määramiseks leiti õhukindlate anumate siseruumala, millest lahutati nendes paiknevate komponentide ruumala. CAD programmid võimaldavad jõuda võrdlemisi täpse lõpptulemuseni, kuid allveesõiduki laadsete süsteemide edasi arendamine eeldab katse-eksituse meetodi rakendamist ning lõplik peenhäälestamine tehakse alati vees.

### **3.1.7. Mehaaniline tugevus ja vastupidavus hüdrostaatilisele rõhule**

Allveesõidukil paiknevate mahutite mehaanilise vastupidavuse hindamiseks analüüsiti plastikust konstrueeritud detaile Siemensi simulatsioonitarkvaras Femap, rakendades lõplike elementide meetodit (LEM). Selle käigus jaotatakse keeruka kujundusega keha, mille füüsiliste vastupidavuse hindamiseks puuduvad konkreetsete matemaatilised valemid, väiksemateks ja lihtsamateks kujunditeks või elementideks, mida osatakse matemaatiliselt lahendada. See võimaldab analüüsida, kuidas mõjuvad keeruka konstruktsiooniga kehadele erinevad välised jõud. [7] LEM oli eelistatud lahendusviisiks, sest see võimaldab arvestada ka sellega, kuidas kolmemõõtmelises kehas erisuundades mõjuvad jõud üksteist mõjutavad. Lihtsamates tugevusarvutustest vaadeldakse, kuidas teatud jõud keha ainult mingis kindlas suunas mõjutab.

Õpilaslitsentsi jaoks on Femap keskkonnas kasutatav elementide sõlmpunktide arv piiratud 32000 ühikuga, mistõttu analüüsiti igast mudelist ainult ¼-ndikku. Sellist lähenemist võimaldab see, et kõik simulatsioonis kasutatud kehad on sümmeetrilised. Simuleeriti olukorda, kus mahuti asub vee all ja talle mõjub 490 kPa suurune hüdrostaatiline rõhk.

Selleks:

1. Detaili lihtsustamine parema elementvõrgu saavutamiseks.
2. Genereeriti detaili geomeetriaile vastav kolmemõõtmeliste elementide võrgustik, kasutades 4-sõlmpunktiga kahemõõtmelisi elemente.
3. Määrati kooste detailide omavahelised kontaktpinnad ja liitetüübid.
4. Defineeriti sümmeetrilised piirangud.
5. Määrati mõjuva jõu suurus, tüüp ning välispinnad, kuhu ümbritsev vesi jõudu avaldab.
6. Lasti programmil arvutada detailile erisuundades mõjuvad summaarsed pinged ja koormused ehk ekvivalentpinged (Von Mises pinge) lineaarstaatilisel meetodil.

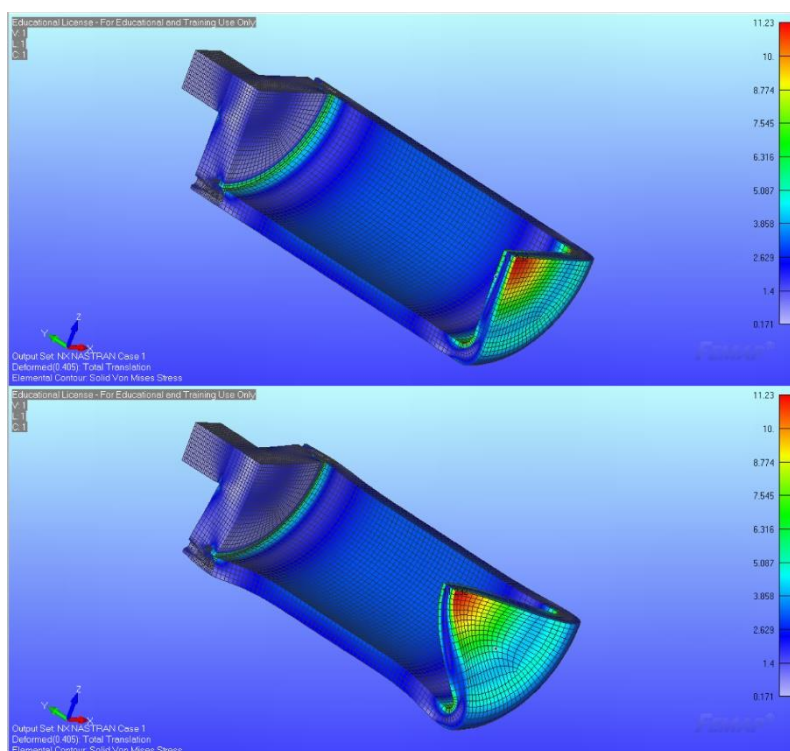
Selleks, et hinnata, kas plastsest materjalist keha peab vastu talle mõjuvale jõule, ilma ise seejuures pöördumatult deformeerumata, saab võrrelda LEM tugevusarvutuste maksimaalsete ekvivalentpingete väärtuseid keha materjali voolavuspiiriga - keha deformeerub, kui ekvivalentpinge ületab materjali voolavuspiiri. LEM analüüs töö põhjalikult välja piirkonnad, kus avalduvad suuremad kohtpinged ning kus ja kuidas deformatsioonid toimuvad. Lineaarstaatilise analüüsi puhul on keha struktuuris toimuvad dislokatsioonid proportsionaalsed kehale rakendatud koormusega, mistõttu on antud meetod rakendatav ja tulemused usaldusväärsed ainult juhul, kui keha struktuuris toimuvad nihked on väikesed. Kui lineaarstaatilise analüüsi tulemustes on struktuuraalsed nihked suuremad kui kehale loodud elementvõrgu üksiku elemendi läbimõõt, tuleb olukorda käsitleda mittelineaarsena ja rakendada dünaamilist laadi simulatsioon. [8, 9, 10]

Disainitud allveesõiduki välimist korpust ei analüüsitud, sest see peab maksimaalsel töösügavusel mõjuvale rõhule ilma probleemideta vastu. Selle pinda ümbritseb mõlemalt poolt vesi, mistõttu sisemuses olev vesi vastusurvestab korpusele väljaspoolt mõjuvat rõhku nii, et jõud võrdsustuvad.



### 3.1.7.1. Akupaki analüüs

Akupakk on 4 mm seinapaksusega ja ümara profiiliga. Simulatsiooni tulemused kinnitavad, et disainitud akupaki korpus on ettenähtud töösügavusel töötamiseks piisavalt tugev. 50 meetri sügavusel vee all tekib akupaki korpuses maksimaalne ekvivalentpinge suurusega 11.23 MPa, mis on ligikaudu 2 väiksem kui polüstüreeni voolavuspiir 22 Mpa. Suurima pingekonsentratsiooniga ala asub keset korpuse põhja, kus plastik paindub rõhu mõjul 0.405 mm võrra (joonis 3.22).

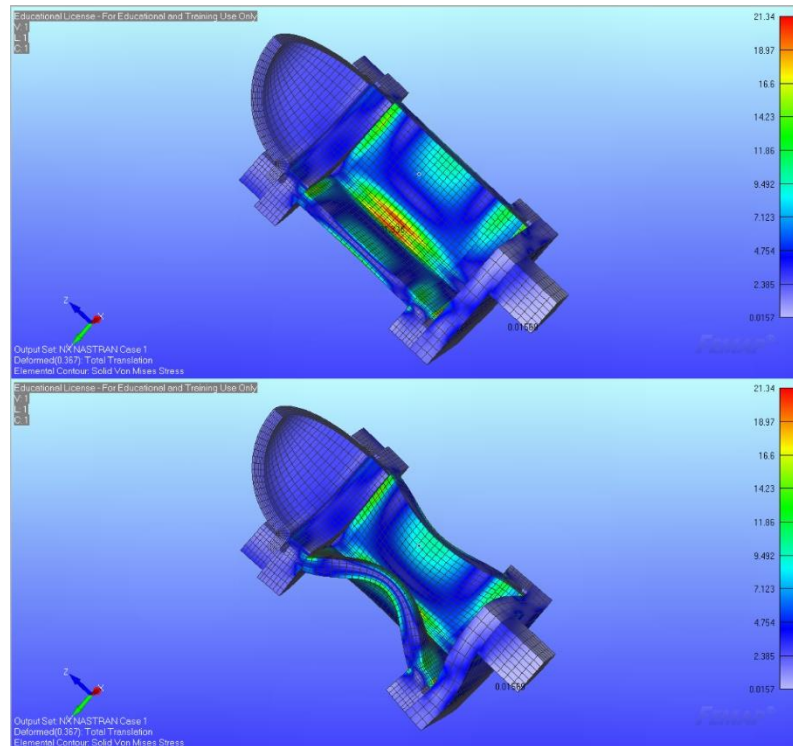


**Joonis 3.22.** Femap simulatsiooni arvutatud ekvivalentpinged akupaki korpuses 490 kPa suuruse hüdrostaatilise rõhu mõjul. Alumine joonis näitab tegelike deformatsioonide 50-kordset võimendust.

### 3.1.7.2. Kaamerakambri analüüs

Kaamerakamber on 3 mm seinapaksusega ja kandilise profiiliga. Simulatsiooni tulemused kinnitavad, et disainitud kaamerakambri korpus on ettenähtud töösügavusel töötamiseks piisavalt tugev. 50 meetri sügavusel vee all tekib korpuses maksimaalne ekvivalentpinge suurusega 21.34 MPa, mis on ligilähedane polüstüreeni voolavuspiirile 22 Mpa. Suurima

pingekonsentratsiooniga ala asub korpuse nurkades, kus plastik paindub rõhu mõjul 0.367 mm võrra (joonis 3.23).



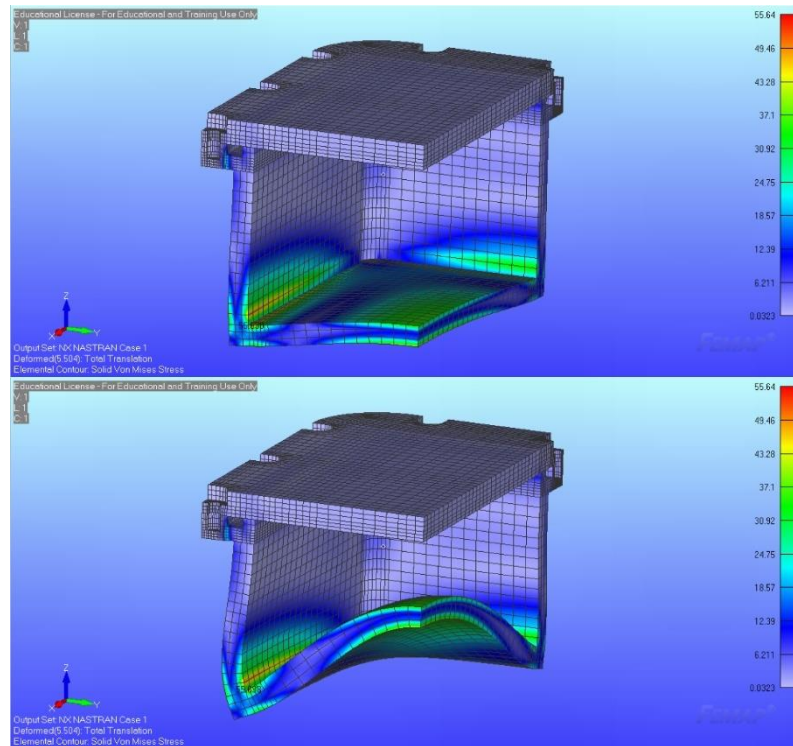
**Joonis 3.23.** Femap simulatsiooni arvutatud ekvivalentpinged kaamerakambri korpuses 490 kPa suuruse hüdrostaatilise rõhu mõjul. Alumine joonis näitab tegelike deformatsioonide 30-kordset võimendust.

Antud detaili vastupidavust oleks võimalik lihtsalt tõsta, tehes kaamerakambri korpuse välisnurgad ümaraks.

### 3.1.7.3. Elektroonikakambri analüüs

Elektroonikakamber on 5 mm seinapaksusega ja kandilise profiiliga. Simulatsiooni tulemused kinnitavad, et disainitud elektroonikakambri korpus ei ole ettenähtud töösügavusel töötamiseks piisavalt tugev. 50 meetri sügavusel vee all tekib korpuses maksimaalne ekvivalentpinge suurusega 55.64 MPa, mis on rohkem kui kaks korda suurem

kui polüstüreeni voolavuspiir 22 Mpa. Suurima pingekonsentratsiooniga ala asub korpuse põhja küljenurkade keskosas (joonis 3.24).



**Joonis 3.24.** Femap simulatsiooni arvutatud ekvivalentpinged elektroonikakambri korpuses 490 kPa suuruse hüdostaatilise rõhu mõjul. Alumine joonis näitab tegelike deformatsioonide 5-kordset võimendust.

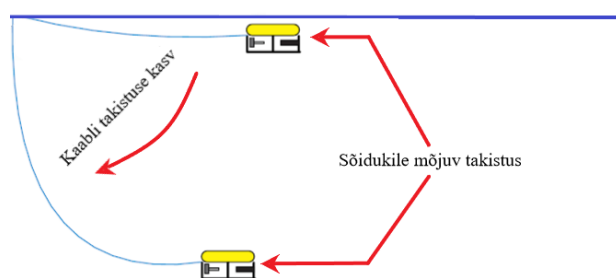
Kuigi on kindel, et antud keha 490 kPa suurusele rõhule vastu ei pea, ei ole staatilise analüüsi väljastatud ekvivalentpinge enam tõne, sest materjali voolavuspiir on ületatud. Saadud tulemuste parandamiseks tuleb elektroonikakambri põhja lisada toestusi või kasutada tugevamat materjali.

### 3.1.8. Allveesõiduki hüdrodünaamiline analüüs

Õppelitsentsiga Femap simulatsioonitarkvara piirangute tõttu puudus tarkvara kus detailset hüdrodünaamilist analüüsi läbi viia. Seetõttu arvutati allveesõidukile ja tema kaablile vees liikumisel mõjuv takistus Mathcad töölehel. Arvutused tehti allveesõiduki liikumiskiirusel

0.5 ja 1 m/s, eeldusel, et kaablit on vees 50 meetrit ja ta jookseb otse alla, ehk on sõidusuunaga täiesti risti. 50 meetri sügavusel ja 0.5 m/s liikumiskiiruse juures mõjub sõiduki kaablile vastu 22.125 N ja 1 m/s liikumisel 88.5 N suurune jõud. Allveesõiduki liikumisel on vastu mõjuvad jõud vastavalt 2.273 N ja 9.092 N.

Arvutustest on näha, et allveesõiduki geomeetria optimeerimisel pole kõige suurem mõju seadme jõudlusele, sest valdava osa süsteemile mõjuvast takistusest tekib allveesõiduki kaablil. Suurendades sõiduki kiirust poole võrra, kasvab allveesõidukile ja tema kaablile mõjuv takistus 4 korda. Töösügavuse kasvades ületab kaablile mõjuv takistus peagi sõidukile mõjuva takistuse, mille suurus on töösügavusest praktiliselt sõltumatu (joonis 3.25).

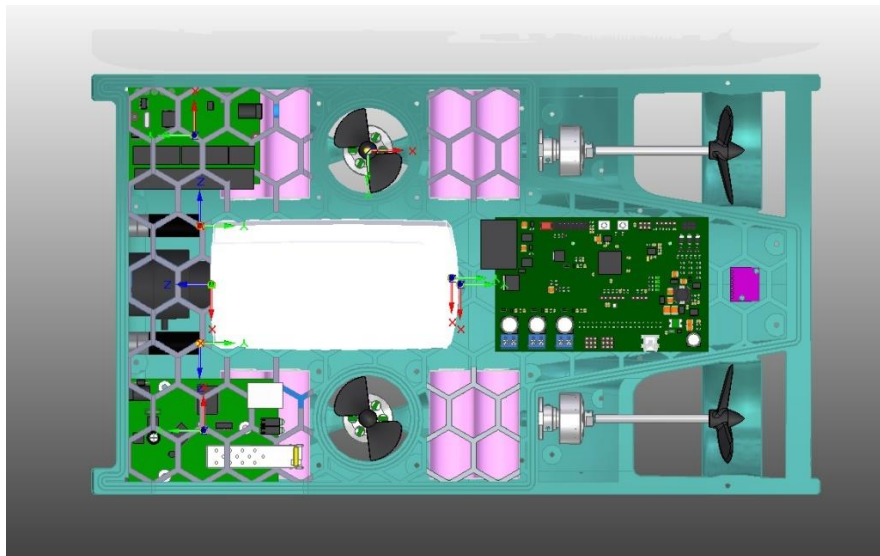


**Joonis 3.25.** Allveesõidukile ja kaablile mõjuv takistus sõltuvalt töösügavusest [1].

Ka kaabli takistust oleks võimalik märgatavalt vähendada, kui selle profiili hüdrodünaamilisemaks muuta. Selleks kasutatakse allvee kaablitel süneetilistest kiust narmaid või vahtplastist ümbriseid [1].

### 3.1.9. Disaini progress

Kogu projekti vältel on allveesõiduki disaini ja mehaanilist ülesehitust lugematul arvul kordi analüüsitud ning täiendatud. Selle tulemusena on esimese prototüübi disain algupärasega võrreldes drastiliselt muutunud. Esimese allveesõiduki prototüübi aluskorpus ja komponendid on näidatud joonisel 3.26.



**Joonis 3.26.** Algupärane allveesõiduki disain koos komponentidega.

### **3.2. Kondensatsioon**

Liigniiskuse eemaldamiseks kasutatakse õhku täis mahutites taaskasutatavaid silikageeli absorbent graanuleid.

## **4. SEADME KOMPONENTIDE KIRJELDUS**

Käesolevas peatükis tutvustatakse projekteeritud allveesõiduki juurde kuuluvaid komponente ja nende omadusi.

### **4.1. Propulsiivseade**

#### **4.1.1. Mootorid**

Väikeste vaatlusklassi allveesõidukite veojõu süsteemid on peaaegu eranditult elektrilised. Suuremate masinate puhul kasutatakse lisaks hüdraulilisi lahendusi, mis aga komponentide rohkuse tõttu jäävad väiksematele ebapraktiliseks. Peamiselt kasutatakse harjadeta alalisvoolumootoreid, mis on [1]:

- võimsad;
- väikesed;
- suure kasuteguriga;
- lihtsa ehitusega (töökindlad);
- väga täpselt juhitud;
- odavad.

Samas on nende kasutamisel allveesõiduki mootorina ka mitmed puudused [1]:

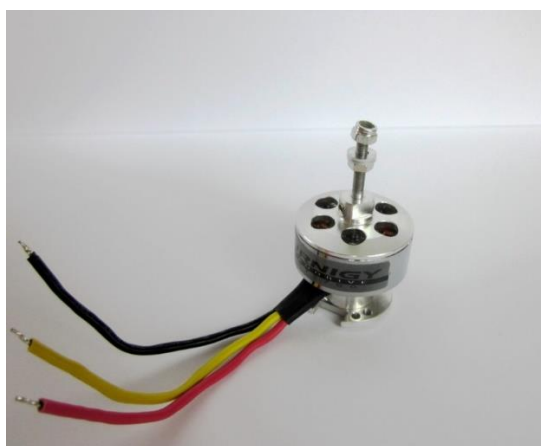
- tarbivad palju energiat;
- täpseks juhtimiseks keerulised kontrollida;
- kõrge pöörlemiskiirus.

Sobiva alalisvoolumootori valimisel võeti arvesse mitmeid mootori parameetreid nagu nimipinge, voolutarve, suurus ja Kv markeering. Lisaks lähtuti teiste allveesõidukite ehitajate kogemustest ja soovitudest. Keha liikuma panemise seisukohalt on kõige olulisem tõukejõu ja keha liikumisel veekeskkonna poolt mõjuva takistuse suhe. Seega mootorite

valimisel on peamine, et nende poolt toodetav summaarne tõukejõud ületaks kogu allveesõiduki korpusele ja selle kaablile mõjuva takistuse.

Otsiti võimalikult väikest ja madala voolutarbega mootorit, mille nimipinge vahemik pidi sobima toiteallika 10.8V väljundpingega. Kuna pöörlemiskiiruse kasvades pöördemoment väheneb on allveesõiduki puhul optimaalne, et mootor töötaks võrdlemisi madalatel pööretel, tagades suurema pöördemomendi [1]. Suurem pöördemoment on allveesõiduki juures ihaldusväärne, sest see annab veevoolus parema kohalt liikumise võime (kiirenduse). Kuna lihtsama süsteemi ja mootorite efektiivsuse säilimise huvides ei soovitud kasutada reduktoreid mootorite pöörlemiskiiruse langetamiseks ja suurema pöördemomendi saavutamiseks, valiti võimalikult madala Kv markeeringuga mootorid.

Välja valitud mootoriteks on Turnigy Aerodrive DST-700 harjadeta alalisvoolumootorid (joonis 4.1).



**Joonis 4.1.** Turnigy Aerodrive DST-700 harjadeta alalisvoolumootor.

Tehniline spetsifikatsioon:

- Kv: 700 p/V;
- nimipinge: 7.4-14.4 V;
- maksimaalne vool: 11 A;
- võimsus: 140 W;
- kaal: 65 g;

- võlli läbimõõt: 3 mm.

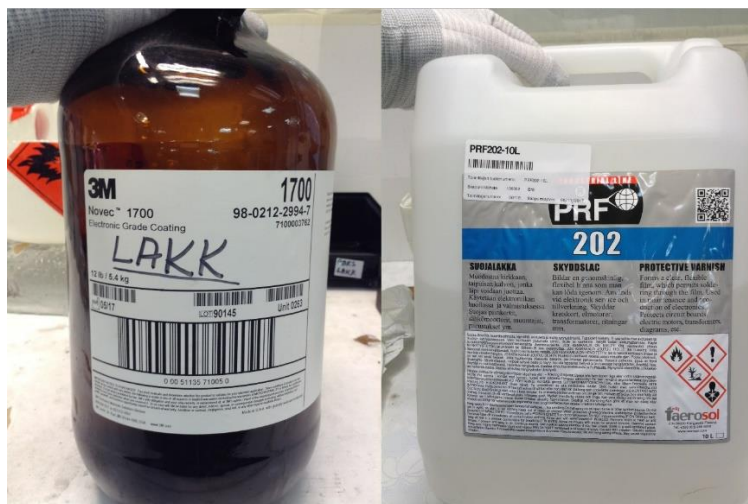
DST-700 mootor on hobikasutajate seas laialdaselt testitud ning oma töökindlust tõestanud. Sama mootorit kasutab oma allveesõidukites ka USA idufirma OpenROV.

Antud mootori käitamiseks on tarvis elektroonilist kiiruse kontrolleri (ingl k *electronic speed controller*, ESC). ESC võimaldab täpset ja mõlemasuunalist kontrolli mootori üle, kuid seejuures vajab sisendsignaaliiks pidevat pulsilaiusmodulatsiooni (ingl k *pulse with modulation*, PWM) mikrokontrolleri digitaalväljundist. Kuigi antud mootorid pole loodud veekeskkonnas töötama, saab neid seal suurepäraselt rakendada. Mootoreid vees kasutades tekib kaks peamist probleemi:

1. Elektrilühise tekke oht;
  2. Korrosioon.
1. Eeldusel, et mootorite juhtmete jätkud on tehtud vettpidavaks, puudub lühise tekke oht ja ka otsene vajadus mootoreid vee juurdepääsu eest kaitsta, sest antud mootorite mähisetraadid on tugeva emailkattega isoleeritud.
  2. Mootorite metallist osade korrodeerumine ilma mootoreid vettpidavaks tegemata on pikas perspektiivis möödapääsmatu. Kõige enam mõjutab see kuullaagrite tööd, mis võib tõsiselt kahandada mootori jõudlust. Üheks võimaluseks mootorite eluiga pikendada on asendada tavalised laagrid keraamiliste, roostevabast terasest või plastikust laagritega. Teiseks saab metallist osad katta emaili, laki või silikoonil baseeruva määrdega. Siiski ka järjepidev loputamine ja määrimine pärast igakordset kasutust tagab juba võrdlemisi pika mootori kuullaagrite eluea.

Mootorite vastupidavuse tõstmiseks kaeti staatori mähised ja rootori magnetid kolmekordselt nii M3 Novec 1700 kui ka PRF202 lakikihihiga (joonis 4.2).





**Joonis 4.2.** Mootorite mähistele ja magnetitele täiendavaks kaitseks pandud lakid M3 Novec 1700 ja PRF202.

Lisaks asendati metallist kuullaagrid plastikust (POM) laagritega (joonis 4.3).



**Joonis 4.3.** Allveesõiduki mootorite algupärased metallist ja uued plastikust kuullaagrid.

Juhul, kui eelnimetatu osutub ebapiisavaks, võib hiljem rakendada täiendavaid meetodeid nagu mähiste valamine epoksüvaiku, paigutada mootorid suletud korpustesse või kasutada magnetilise sidestusega ülekannet. Siiski, arvestades kasutatud mootori madalat tükihinda (7 eurot), ei õigusta kulukamad ning mitmeid lisaosi nõudvad lahendused ennast ära - riknemisel saab mootori suuremate kuludeta välja vahetada.

#### 4.1.2. Sõukruvid

Selleks, et rakendada mootoreid allveesõiduki edasiliikumiseks vajaliku jõu tekitamiseks, kasutatakse sõukruvisid. Õige sõukruvi valik on oluline tagamaks võimalikult suur allveesõiduki jõudlus. Optimaalse sõukruvi korral [1]:

- väheneb oluliselt mootorite energiatarve;
- pikeneb akude kestvus;
- paranevad mootorite juhtahela termilised näitajad;
- suureneb genereeritav tõukejõud.

Valitava propelleri diameeter, samm, labade arv ja kuju sõltub eelkõige kasutuseesmärgist. Üldised põhimõtted väiksemate sõukruvide valikul [11]:

- mida vähem labasid, seda parem – labade arvu kasvuga langeb sõukruvi tootlikkus. Samas langeb labade vähendamisega sõukruvi stabiilsus, tasakaal ja tõuseb vibratsioon ning kavitatsioon. Tihtipeale on optimaalne 3-labaga sõukruvi;
- mida kitsamad labad, seda parem – väiksem takistus;
- mida suurem diameeter, seda parem – suurem vett liigutav pind;
- mida väiksem samm, seda parem;
- mida jäigemad labad, seda parem – pehmed labad painduvad ning seeläbi kaotatakse efektiivsuses.

Tihtipeale leitakse mingiks kindlaks kasutuseks parim mootori ja sõukruvi kombinatsioon praktilisel teel katsetades, sest teoreetiliselt on seda keeruline hinnata [1]. Parima tulemuse annab sõukruvi, mis võimalikult madala pöördemomendi ja voolutarbe juures annab kõige rohkem tõukejõudu.

Pikiteljelistele mootoritele valiti 65 mm läbimõõduga 3-labalised ja vertikaalsele 50 mm läbimõõduga 2-labaline sõukruvi firmalt Graupner (joonis 4.4).



**Joonis 4.4.** Allveesõiduki mootoritel kasutatavad sõukruvid: 1 – Graupner 2308.65; 2 – Graupner 2303.50.

Tüür- ja pakpooi sõukruvid on vastupidiste sammudega, et nad üksteise suhtes erinevas suunas liiguksid ja seeläbi võimalikult sirgjoonelise liikumise tagaksid. Vertikaalse mootori sõukruvi on 2-labaline, sest see on oma põhi pöördesuunale vastupidi pööreldes mõnevõrra efektiivsem kui 3-labalised sõukruvid. Vertikaalsuunalise liikumise juures on kriitiline, et nii laskudes kui ka tõustes oleks võimalikult palju tõukejõudu. Vastasel juhul võib juhtuda, et põhjast ei suudeta suure takistuse tõttu pinnale tõusta. 3-labalised sõukruvid on ühte pidi pööreldes tunduvalt efektiivsemad kui teist pidi, sest horisontaalsete mootorite puhul on tähtis eelkõige edaspidi liikumine. Sõukruvide valikul lähtuti eelkõige teiste allvesõidukite ehitajate katsetuste tulemustest.

#### **4.1.3. Düüs**

Selleks, et suurendada käitursüsteemi poolt tekitatavat tõukejõudu ja vähendada voolutarvet, kasutati allvesõiduki sõukruvide ümber düüse (joonis 4.5).



**Joonis 4.5.** 3D prinditud allveesõiduki düüs.

Düüsi ülesandeks on juhtida sõukruvi peale nii palju vett kui võimalik ja seda eelkõige madalatel kiirustel, kui on tarvis maksimaalset tõukejõudu. Sõukruvi pöörlemisel tekib selle labade taha kõrge rõhuga piirkond ja labade ette madala rõhuga piirkond. Tulemuseks on suur rõhkude vahe, mis surub vee joana ühelt sõukruvi laba poolelt teisele, tekitades veesõidukit edasiviiva jõu. Ainult sõukruvi kasutades tekivad vee liikumisel labade tippudes suured kaod ning märkimisväärne osa tõukejõust hajub ümbruskonda laiali. Düüs moodustab sõukruvi ümber toru, mille tulemusena enamik kadudest suunatakse otse sõukruvi ette sõidukit edasi tõukavaks jõuks. Lisaks pakub düüs sõukruvile ka täiendavat kaitset. Suurtel kiirustel pole düüsid praktilised, sest düüsi profiilile mõjuv takistus ületab düüsiga saavutatud lisand tõukejõu. [2, 4, 12] Arvestada tuleb ka, et kõik sõukruvid ei pruugi düüsidega sobida ning võivad kaasa tuua jõudluse languse. Allveesõiduki düüsidel kasutati Rice profiili, mida kohandati, et luua düüs siseläbimõõduga 68 mm ja kõrgusega 30 mm.

#### **4.1.4. Elektroonilised kiiruse kontrollid**

Harjavaba alalisvoolumootori juhtimiseks kasutatakse 12 A markeeringuga AfroESC programmeeritavat elektroonilist kiiruse kontrolleri (joonis 4.6).



**Joonis 4.6.** 12 A AfroESC elektrooniline kiirusekontroller.

Kiiruse kontroller genereerib vastavalt sisendsignaalile mootorile madalapingelist kolmefaasilist vahelduvvoolu, arvestades seejuures rootori asukohaga [13]. Kiirusekontrollerid programmeeriti reageerima sissetulevale 50 Hz PWM signaalile täiteteguriga 5 kuni 10 %, ehk 20 ms perioodiga signaalile, kus impulsid on pikkusega 1 kuni 2 ms. Antud vahemikus tähistab 1.475 kuni 1.525 ms seisusignaali ning 1 ja 2 ms maksimaalset kiirust eri pöörlemissuundades.

#### **4.1.5. Käitursüsteemi test**

Allveesõiduki mootorikomplekti tööd imiteeriti agregaadis, kus kangiga liigendvõlli külge monteeritud mootorikomplekt asetati veega täidetud mahutisse ning pandi tõmbama digitaalkaalu (joonis 4.7).

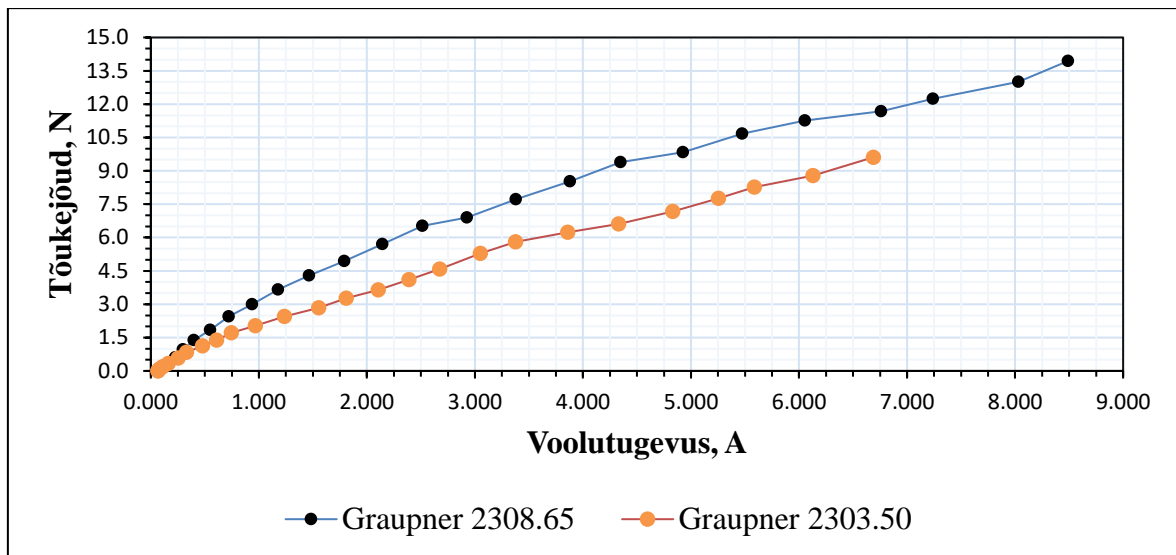


**Joonis 4.7.** Mootorikomplekti agregaat ja testplatvorm.

Katse käigus mõõdeti konstantse 10.8 V toitepinge juures mootori erinevatel töökiirustel voolutarvet ja tekitatavat tõukejõudu, mis on proportsionaalne mootori poolt vees paiknevale allveesõidukile tekitatava tõukejõuga.

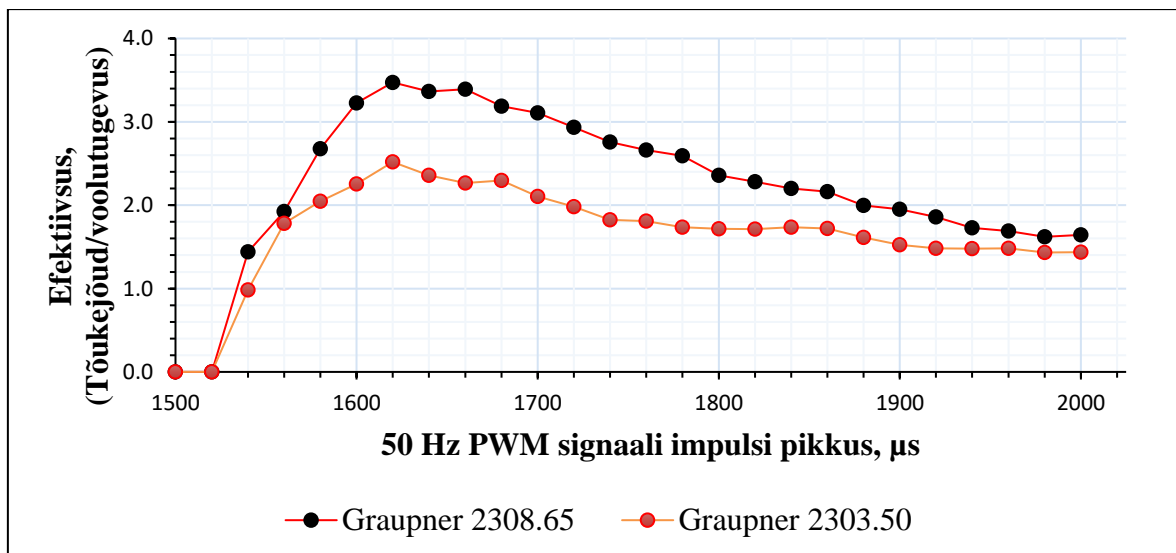
Tulemused kinnitavad, et horisontaalselt paiknevatele mootoritele valitud 3-labalised sõukruvid on terves töövahemikus põhisuunda pidid pööreldes ligikaudu kaks korda efektiivsemad. Vertikaalse mootori 2-labalise sõukruvi puhul on pöörlemissuundade väljundi vahe väiksem, jäädes 25% juurde. Lisaks selgus, et valitud propellerid ei ole düüsisiga kasutamiseks optimaalsed, sest annavad ilma düüsisita sama voolutarbe juures kuni 25% rohkem tõukejõudu.

3-labalise sõukruviga tarbib mootor mööda sõukruvi põhisuunda pööreldes maksimaalselt 8.49 A voolu ja annab 13.94 N tõukejõudu. 2-labalise sõukruviga tarbib mootor mööda sõukruvi põhisuunda pööreldes maksimaalselt 6.69 A voolu ja annab 9.61 N tõukejõudu (joonis 4.8).



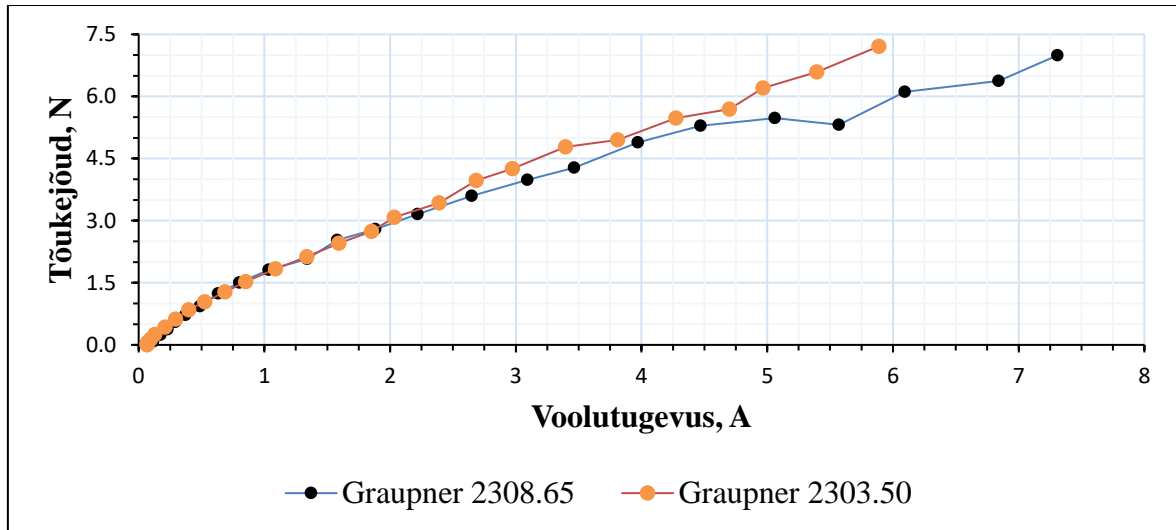
**Joonis 4.8.** Turnigy Aerodrive DST-700 mootori voolutarbe ja tõukejõu karakteristik Graupner 2308.65 3-labalise ja 2-labalise Graupner 2303.50 sõukruviga, pööreldes mööda sõukruvi põhisuunda.

Mõlemal juhul saavutab mootor oma maksimaalse efektiivsuse mikrokontrolleri PWM väljundsignaali 1620  $\mu\text{s}$  pikkuse impulsi juures (joonis 4.9).



**Joonis 4.9.** Turnigy Aerodrive DST-700 mootori efektiivsus sõltuvalt mikrokontrolleri 50 Hz PWM väljundsignaali täidetegurist, kasutades mööda sõukruvi põhisuunda pöörlevat Graupner 2308.65 3-labalist ja Graupner 2303.50 2-labalist sõukruvi.

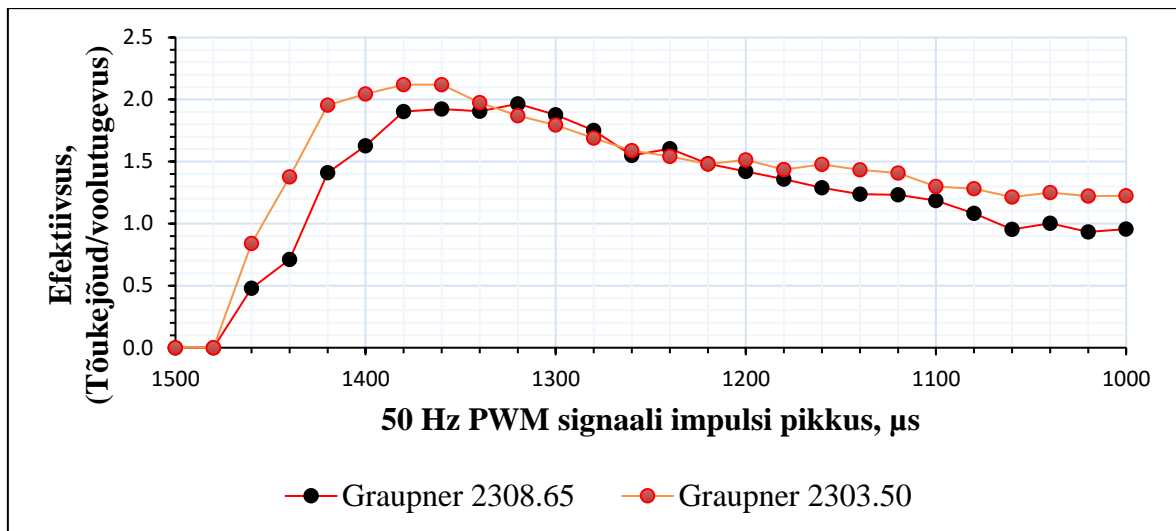
Vastu sõukruvi põhisuunda pööreldes tarbib mootor 3-labalise sõukruviga maksimaalselt 7.31 A voolu ja annab 7 N tõukejõudu. 2-labalise sõukruviga tarbib mootor vastu sõukruvi põhisuunda pööreldes maksimaalselt 6.69 A voolu ja annab 9.61 N tõukejõudu (joonis 4.10).



**Joonis 4.10.** Turnigy Aerodrive DST-700 mootori voolutarbe ja tõukejõu karakteristik Graupner 2308.65 3-labalise ja Graupner 2303.50 2-labalise sõukruviga, pööreldes vastu sõukruvi põhisuunda.

Vastupidisel pöördesuunal saavutab mootor 3-labalise sõukruviga oma maksimaalse efektiivsuse mikrokontrolleri PWM väljundsignaali 1320  $\mu$ s pikkuse impulsi juures. 2-labalise sõukruviga on efektiivseim punkt 1360  $\mu$ s pikkuse signaali juures (joonis 4.11).



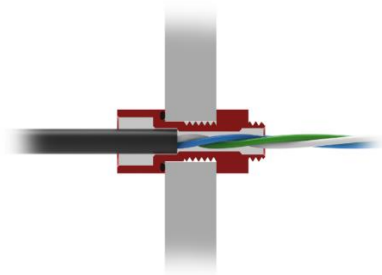


**Joonis 4.11.** Turnigy Aerodrive DST-700 mootori efektiivsus sõltuvalt mikrokontrolleri 50 Hz PWM väljundsignaali täietegurist, kasutades vastu sõukruvi põhisuunda pöörlevat Graupner 2308.65 3-labalist sõukruvi.

Kõik eelpool esitatud tulemused on katsest, kus kasutati sõukruvisid ilma düüsita. Selleks, et vältida sõukruvi valikust tingitud kadusid, tuleb allveesõidukilt düüsid eemaldada või leida katse ja eksituse meetodil düüsiga kasutamiseks sobiv sõukruvi.

## 4.2. Läbiviigud

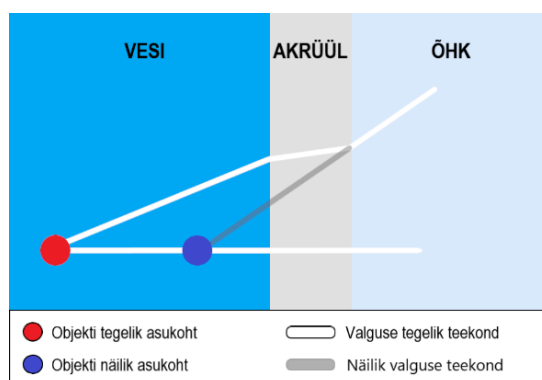
Selleks, et vedada juhtmed õhukindlatesse korpustesse kasutatakse o-rõngastihendiga tihendatud läbipuuritud polte. Vettpidava isolatsiooni saavutamiseks valatakse läbiviigu sisse veekindlat epoksüliimi Loctite EA 3430 (joonis 4.12).



**Joonis 4.12.** Veekindel juhtme läbiviik [14].

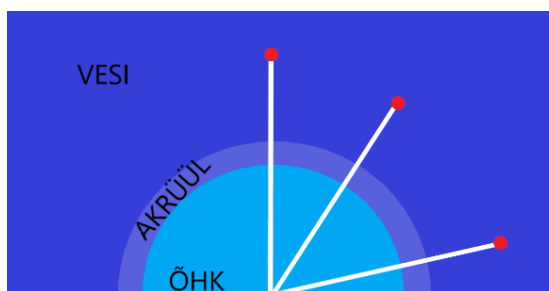
### 4.3. Kaamerakambri aken

Valguskiired levivad sirgjooneliselt ainult ühtlases keskkonnas. Jõudes mingi teise keskkonnani, muudavad kiired oma suunda - murduvad, peegelduvad ja hajuvad. Vaadates vee all mingit objekti läbi lameda ja korrigeerimata läätse, põhjustab valguskiirte murdumine moonutusi ning viga fookuses ja vaateväljas. Selle mõjul tekib kaamera ette moonutatud äärtega virtuaalne pilt ning kõik veealused objektid tunduvad meile ligi veerandi võrra lähemal ja suurematena, kui nad tegelikult on. Sellest tingituna on ka kaamera vaateväli tunduvalt kitsam kui õhus. [1] Enne allvesõiduki kaamera läätseni jõudmist läbib valgus kolm meediumit – vesi, akrüülist aken ja kaamerakambris olev õhk (joonis 4.13).



**Joonis 4.13.** Valguskiire levik läbi lameda akrüülist läätse.

Eelpool mainitud probleemide leevendamiseks on kaamerakambri aknaks kasutatud kumerat akrüülist läätse. Kumer pind kõrvaldab moonutused, sest valguskiired lähevad murdumata läbi akrüüli (joonis 4.14) [15].



**Joonis 4.14.** Valguskiirte levik läbi kumera läätse.

Lisaks talub kumer pind paremini rõhku. Paigaldatud läätse maksimaalne töösügavus on 1000 m.

#### **4.4. Sidevahendid**

Järgnevatel põhjustel otsustati allveesõiduki ja maapealse kasutajaliidese vahel kasutada kiudoptilist ühendust [1]:

- turvaline – vees objektide taha takerduda võiv kaabel ning pidevalt kokku ja lahti käivad ühendused on rikkealtid kohad. Valguskaabel tagab galvaanilise eralduse, sest võimaldab hoiduda elektrijuhtidest koosneva kaabli kasutamisest, mis välistab elektrilühise tekkimise ohu kaabli või pistiku rikke korral;
- võimaldab „märga“ pistikute kokku ja lahti ühendamist;
- saab kasutada kuni mitme kilomeetri pikkust ühenduskaablit;
- suure ribalaiusega;
- vähendab oluliselt ühenduskaabli läbimõõtu ning seeläbi allveesõidukile mõjuvat takistust ja energiakulu.

##### **4.4.1. Meediakonverter**

Kasutatakse kahte Etherneti kohtvõrku kiudoptiliseks meediaks muundavat meediakonverterit, millest üks ühendub juhtliidese arvutiga ja teine allveesõiduki juhtüksusega Raspberry Pi. Valitud meediakonverterid toetavad kuni ühe gigabitist Etherneti ning rakendavad monomoodilist ühekiulist andmeedastust (joonis 4.15).



**Joonis 4.15.** Etherneti signaali valgussignaalsiks tegev meediakonverter.

Allveesõiduki elektroonikakambris paiknev üksus on kaalu ja ruumi kokkuhoidmiseks oma korpusest eemaldatud.

#### **4.4.2. Kaabel**

Andmevahetus allveesõiduki laadsete süsteemidega toimub alati läbi mingisuguse kaabli, sest kõrgesageduslikud raadiolained levivad vees ainult mõne sentimeetri sügavusele. Väga madalasageduslikud raadiolained levivad kaugemale, kuid pole videopildi edastamiseks kaugeltki piisava ribalaiusega. [1]

Suurtel süvavee allveesõidukitel kasutatakse jämedaid kaableid, kuhu on integreeritud vaskjuhtmed koos valguskaabli optiliste kiududega - elektrijuhte kasutatakse toitevoolu ülekandmiseks ning valguskaablit andmevahetuseks. Väiksemad vaatlusklassi allveesõidukid kasutavad üldjuhul kuni 1 cm läbimõõduga kaablit, kus sõltuvalt andmeedastustehnoloogiast on 1 kuni 4 keerdpaarjuhet. Levinumad andmevahetusprotokollid on Ethernet ja Homeplug. Allveekaablite tugevdamiseks põimitakse juhtmete vahele mitmesuguseid sünteetilisi kiudusid, millest levinumad on aramiid kiud Kevlar, vedelikkristallpolümeer Vectran ja polübensoksasoolkiud Zylon. Kaabli kiudude ja juhtmete vahele lisatakse ka vettplokeerivaid ühendeid ja segusid. Süvaveekaablitele lisatakse armatuuriks ka terastraate. Kaablite sisu ümbritsetakse kattega, mis seob kõik komponendid kokku ning annab oma madala tihedusega kaablile vees heljuva oleku. Enamlevinuteks katematerjalideks on polüuretaanvaht, polüolefiin või

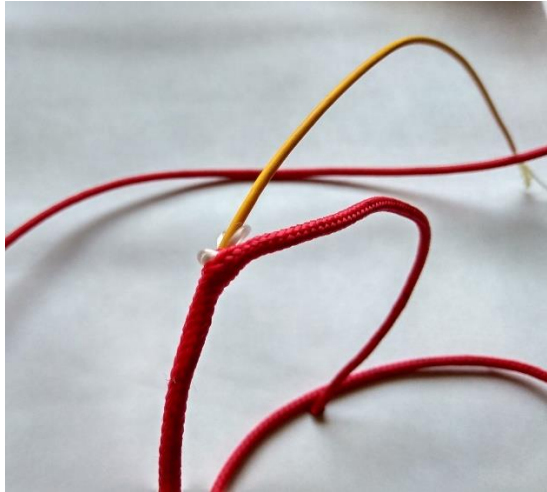
termoplastiline elastomer. [1] Tavaliselt vaatlusklassi allveesõidukite kaableid toitevoolu ülekandmiseks ei kasutata, sest see tõstab kogu süsteemi hinda, komponentide kogust ja keerukust. Lisaks sellele on toitekaabli kasutamine ohtlik, sest kadude vähendamiseks tuleks toitepinge tõsta vähemalt 48 V peale.

Allvee tingimustes ning eriti väikeste vaatlusklassi sõidukite kaablite puhul on üheks kriitilisemaks parameetriks kaabli läbimõõt. Vees paikneva kaabli läbimõõdu ja pikkuse kasvades tõuseb märgatavalt kaablile mõjuv takistus, mis moodustab valdava osa seadmele mõjuvast kogutakistusest. Seepärast tuleks kasutada võimalikult optimaalse tugevuse ja läbimõõdu suhtega kaablit. [1] Allveesõidukite jaoks müüdavad valguskaablid on võrdlemisi kallid (alates 3.5 -eur meeter), mistõttu otsustati kasutada tavalist LC ja FC tüüpi pistikutega ühekiulist monomoodilist valguskaablit (joonis 4.16).



**Joonis 4.16.** Allveesõiduki valguskaabel.

Valitud kaabel on vees uppuv, sest 100 meetrine kaabel läbimõõduga 1.8mm kaalub 348 g, mis annab kaabli keskmiseks tiheduseks  $1368 \text{ kg/m}^3$ . Kaablile heljuva oleku andmiseks ja vastupidavuse tõstmiseks põimiti selle ümber punane 16-st kiust koosnev polüpropüleenist kest koos polüpropüleenist sisuga (joonis 4.17).



**Joonis 4.17.** Allvesõiduki tarbeks vees hõljuvaks tehtud valguskaabel.

Pärast kaabli täiendamist on 100 m valguskaabli läbimõõduks 2.95 mm, kaaluks 664 g ja keskmiseks tiheduseks  $972 \text{ kg/m}^3$ .

Kaabli ühenduspunkti asukoht allvesõidukil peaks olema selle peamise sõidusuunaga võimalikult samasihiline, et vähendada takistust ja liigset koormust pistikust vahetult väljuvale kaabliosale.

## 4.5. Kaamera

Vaatlusklassi allvesõiduki võimalikult kvaliteetne videopilt reaajas on esmatähtis omadus, sest sellel taotlusel sedalaadi allvesõidukeid üldse ehitatakse. Üldiselt on see oluline ka suuremate masinate puhul, sest videopilt annab kasutajale eeldused allvesõidukitega keskkonna vaatlemiseks või töö tegemiseks.

Arendusprotsessi käigus katsetati erinevaid USB ja võrgukaameraid. Kaamerat valiti järgmistel kriteeriumitel:

- pildisensor – peab toetama *Full HD* eraldusvõimet. Vähesel valgusel tõttu tuleks eelistada võimalikult suurt pildisensorit, mille pikslite arv ei oleks ülemäära kõrge. Sellistel sensoritel on suuremad pikslid, mis tagab parema valgustundlikkuse;

- peab olema integreeritud H.264 koodekiga kompressiooni mikrokiip – võtab juhtarvutilt pilditöötlemise koormuse ning vähendab signaaliedastuse andmemahutu;
- kaamera suurus – võimalikult väike ja kerge;
- minimaalne kaadrisagedus – vähemalt 25 kaadrit sekundis;
- USB kaamera korral UVC ühilduv.

Allveesõiduki pardakaameraks valiti Leedus toodetud USB kaamera Kurokesu C1 (joonis 4.18).



**Joonis 4.18.** Allveesõiduki USB pardakaamera Kurokesu C1 koos objektiiviga.

Antud kaameral on 1/3 tolline Aptina AR0330 pildisensor, mis väljastab YUY2/MJPEG/H.264 formaati kodeeritud andmeid ning on UVC ühilduv [16]. Kaameraga kasutatakse 90° vaateväljaga objektiivi.

Üldiselt sobivad võrgukaamerad oma hea valgustundlikkuse ja kõiksuguste pildi seadistamise lisavõimalustega vähese valgusega tingimustesse küllaltki hästi. Sellegipoolest nende kasutamisest loobuti, sest kõik sobivad võrgukaamerad olid kas liialt suured või kallid. Lisaks oleks võrgukaamera kasutamisel süsteemi lisandunud ka kommutaator.

Kindlasti annaks parima tulemuse Ximea, Point Gray või mõne analoogse kaameratootja kõrgtehnoloogiline kaamera. Antud firmad kasutavad oma kaamerates suuri tippklassi pildisensoreid, millel on kõrge valgustundlikkus, eraldusvõime ja kaadrisagedus. Paraku

eeldab nende kaamerate kasutamine väga suure jõudlusega arvuti olemasolu, sest neil kõigil puudub kaamerapõhine H.264 kompressiooni võimalus.

## 4.6. Juhtpult

Allveesõiduki juhtimiseks kasutatakse mängupulti Logitech F310, mille põhilised funktsioonid on välja toodud joonisel 4.19.



**Joonis 4.19.** Allveesõiduki juhtpult Logitech F310 koos põhiliste funktsioonidega.

Puldi puudmisel pole võimalik allveesõidukit liigutada. Tarkvara edasi arendades luuakse ka arvuti klaviatuuri-põhine juhtimisvõimalus.

## 4.7. Akupakk

Allveesõiduk töötab 10.8 V toitepingega, mis saadakse pardal asuvast kahest omavahel rööbiti ühendatud akupakist kogumahtuvusega 86.4 Wh. Kumbki akupakk koosneb kolmest jadamisi ühendatud 21700 suurusega liitiumioonakust ( $\text{LiNiCoMnO}_2$  ehk INR keemial põhinev) nimipingega 3.7 V ja mahutavusega 4 Ah (joonis 4.20).





**Joonis 4.20.** Vapcell INR21700 3.7 V 4 Ah aku.

Antud akupaki mahtuvuse juures on seadme eeldatav tööaeg kuni 1.5 tundi. Akupakk kannatab kuni 30 A suurust katkematut voolutarvet.

#### **4.8. Valgustid**

Puhtas vees neeldub 90% kogu veepinnalt vee alla jõudvast valgusest esimese 10 meetri juures. Valdav osa punasest spektrist kaob juba 5 meetri ja sinisest 10 meetri peal. Kõige sügavamale jõuab siniroheline valgus, millest 30 meetri sügavusel on alles ligikaudu 5 %. Veealust nähtavust vähendab ka tõsiasi, et mitte kõik veepinnale langevast valgusest ei jõua vee alla, sest osa peegeldub veepinnalt tagasi. [1, 4] Loodusliku valguse puudulikkus sügavamal kui mõni meeter loob vajaduse varustada allveesõiduk valgustitega. Tehisvalgus võib abiks olla madalamas vees, aidates välja tuua veealuste objektide tegelikud värvid.

Oluline on, et kasutatud valgustite valguse intensiivsus oleks reguleeritav, sest liiga ere valgus võib põhjustada täiendavat valguse peegeldumist veesambas hõljuvatelt osakestelt ning seeläbi kaamerat pimestada. Tagasipeegelduva valguse hulk sõltub vee värvusest, osakeste hulgast ja ka valgustite asukohast. Mida suurem on osakestele langeva valguse nurk kaameraobjektiivi suhtes, seda vähem valgust nendelt kaameraobjektiivi peegeldub. [1]

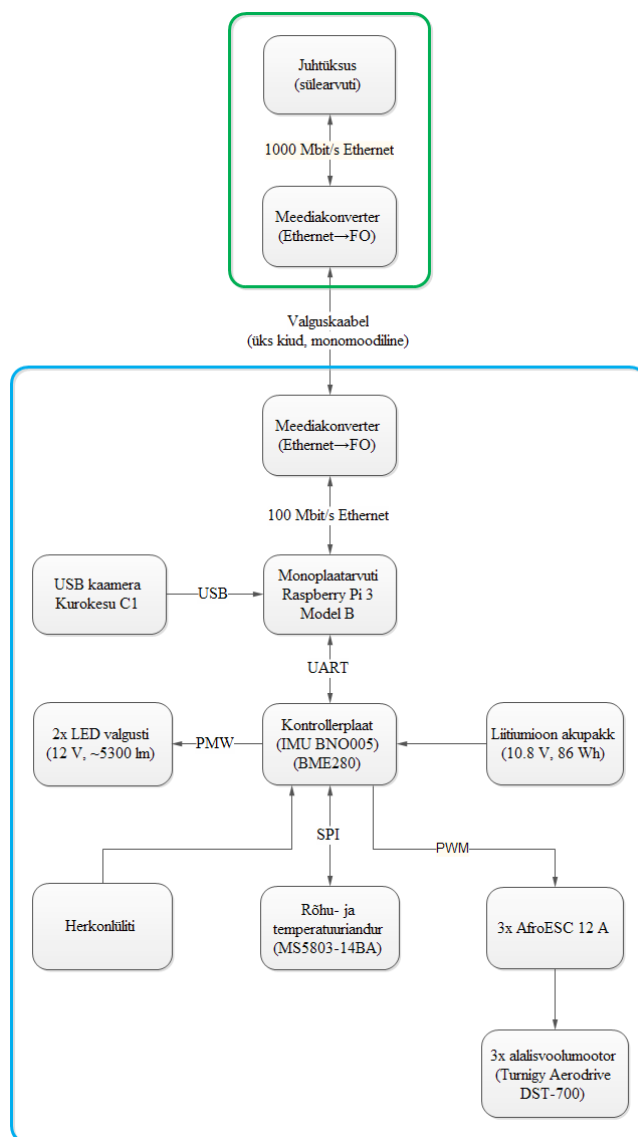
Antud allveesõidukil asub roteeruvas kambris kaks LED valgustit (joonis 4.21), mis on paigaldatud kumbagi äärde, kaamerakambrist võimalikult kaugele. Valgustites kasutatakse Cree XLamp XHP50.2 valgusdiodi nimivõimsusega 18 W ja valgusvooga 2650 lm.



**Joonis 4.21.** Veekindel LED valgusti.

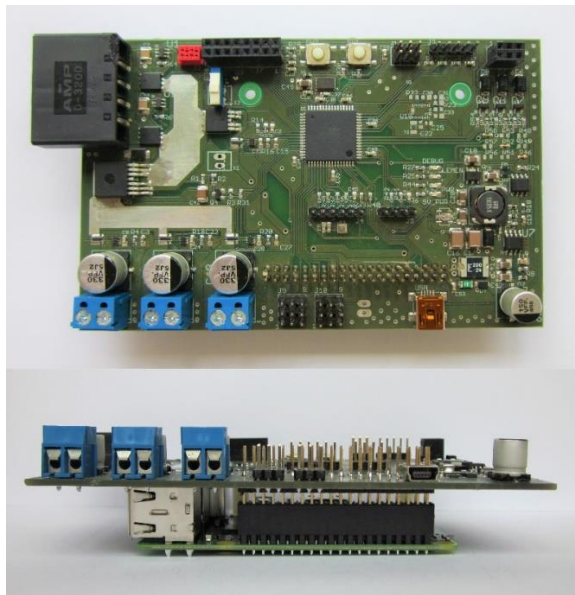
## 5. SEADME RIISTVARALINE DISAIN

Käesolevas peatükis kirjeldatakse lühidalt koostatud kontrolleri trükkplaadi prototüübi disaini ja osasid. Kõik controllerplaadi elektroonikaskeemid ja plaadi kujutised on toodud töö lisas C. Seadme elektroonika topoloogia on nähtav joonisel 5.1.



**Joonis 5.1.** Allveesõidukis kasutatava elektroonika topoloogia.

Kogu seadme keskseks juhtkuseks on ühenduspesaga kontrollerrplaadile kinnituv Raspberry Pi 3 mudel B monoplataarvuti, mis vahendab ja haldab andmeid kasutajaliidese ja kontrolleri vahel (joonis 5.2).



**Joonis 5.2.** Kontrollerrplaat pealt- ja külgsaates koos Raspberry Pi arvutiga.

Raspberry Pi suhtleb kontrollerrplaadiga üle universaalse sünkroonse jadaliidese (USART) ja saab elektritoite otse kontrollerrplaadilt.

## 5.1. Kontrollerrplaat

Allveesõiduki elektritoite, andurite ja mootorite haldamiseks disainiti Microchip Technology AVR seeria ATxmega64A3U mikrokontrolleril põhinev kontrollüksus.

ATxmega64A3U on 8 bitine 3.3 V toitepingel töötav juhtseade, maksimaalse taktsagedusega kuni 32 MHz [17]. ATxmega64A3U valiti tema mitmekülgsede perifeeriaseadmete ja USB liidese olemasolu tõttu. Lisaks oli töö autoril varasem kokkupuude 8 bitiste AVR seeria mikrokontrolleritega. Kontrollerrplaadi disain on teostatud kahekihilisele trükkplaadile tarkvaras Autodesk Eagle.

### 5.1.1. Elektritoite ahelad

Seadme tarbijate elektritoide lülitatakse sisse läbi kontrolleri ühenduses oleva herkonlüli, mis asub valguskaabli pistikupesa vahetus läheduses. Herkonlüli sulgemiseks on allveesõiduki kaabli pistiku külge monteeritud kolm seibikujulist neodüümmagnetit.

Kontrollerplaadi elektritoite sisendites on p-kanaliga väljatransistorid, mis kaitsevad plaati toiteterminalide valesti ühendamise eest. Mootorite elektritoite lülitamist tehakse läbi mikrokontrolleri, mille väljundsignaaliga kontrollitakse madala pingega tüüritavat n-kanaliga väljatransistorit, mille neel on ühendatud p-kanaliga jõu-väljatransistori paisuga. Kõrge signaali korral maandab n-kanaliga väljatransistor jõu-väljatransistori paisu ning avab selle.

Elektritoitesüsteemis tagab Raspberry Pi pardaarvutile stabiilse 5 V toitepinge Texas Instrumentsi integraallülitus TPS54331. TPS54331 on fikseeritud 570 kHz lülitussagedusega madaldav alalispingemuundur väljundvoolu võimekusega 3 A [18].

Mikrokontrolleri ja teiste 3.3 V toitepingel töötavate komponentide käitamiseks kasutatakse kahte lineaarset kuni 1 A väljundvooluga pingestabilisaatorit (LDO) NCP694 firmalt ON Semiconductor. Üks nimetatud pingestabilisaatoritest on ainult andurite ja teiste lisaseadmete käitamiseks, et neid tõrke korral taaskäivitada, ilma mikrokontrollerit välja lülitamata. Programmeerimise lihtsustamiseks on plaadile lisatud USB pesa, mille ühenduse ajal isoleeritakse mikrokontroller põhitoiteallikast, võimaldades programmeerimist ka siis, kui allveesõiduk töötab. USB kaabli külge- ja lahtiühendamisel on Raspberry Pi arvutile tagatud katkestusteta elektritoide.

### 5.1.2. Mõõteahelad

Voolutarbe mõõtmiseks on plaadile paigaldatud neli Texas Instrumentsi voolumõõtjat INA199A1. INA199A1 mõõdab pingelangu, mis tekib elektrivoolu liikumisel läbi madalaoomise takisti. Tekkinud pingelangu võimendatakse sisse integreeritud võimendiga ja edastatakse diskreetimiseks mikrokontrolleri analoog-digitaalmuundurisse. [19]

Iga mootori toiteahelas olev voolumõõtja disainiti mõõtma voolutugevust vahemikus 100 mA kuni 10 A, ülejäänud tarbijate toiteahelas mõõdetakse voolutugevust vahemikus 100 mA

kuni 7 A. Ka 100 milliamprist väiksemad voolutugevused registreeritakse, kuid seljuhul tuleb arvestada rohkem kui 25 % mõõteveaga. Madalaoomiste takistite valikul arvestati voolumõõtja väljundisse ühendatud mikrokontrolleri nimipinge, maksimaalse mõõteahelat läbiv voolutugevuse ja voolumõõtja võimedusteguri (50) suurusega. Voolutugevust mõõdetakse süsteemi toite ja koormuse vahelt (*high side sensing*).

Allveesõiduki aku pinget mõõdetakse takistitest koostatud pingejaguri väljundist.

### 5.1.3. Andurid

Vee all paremini navigeerimiseks, kasutatakse kontrollerplaadile paigaldatud 9-teljelist inertsiaalandurit (IMU) Bosch BNO055. BNO055 on kõrgelt integreeritud mikrokiip, mille sisse kuulub [20]:

- 16-bitine 3-teljeline güroskoop;
- 14-bitine 3-teljeline kiirendusandur;
- 3-teljeline geomagnetiline sensor;
- 32-bitine cortex M0+ mikrokontroller.

Inertsiaalanduriga saab ülevaate allveesõiduki asendist ning liikumissuunast.

Elektroonikakambriist välja paigaldatakse spetsiaalne veesamba rõhu- ja temperatuuri mõõtmise andur MS5803-14BA firmalt TE connectivity. Elektrilühise ära hoidmiseks kaetakse anduri trükkplaat kuni anduri survetundliku pea servani epoksiidliimiga. Antud andur võimaldab mõõta 20 Pa täpsusega kuni 1.4 MPa suurust rõhku, mis võrdub ligikaudu 140 meetri sügavusel vee all mõjuva rõhuga [21]]. Rehkendades rõhuanduri väljundsignaali ümber veesamba sügavuseks, saab määrata allveesõiduki töösügavuse 2 mm täpsusega. Lisaks annab rõhuandur võimaluse luua tarkvaras allveesõiduki automaatse stabiliseerimise ja sügavuse hoidmise funktsioonid.

Lekke tuvastamiseks lisati kontrollerile temperatuuri, rõhu- ja õhuniiskuseandur Bosch BME280, millega on võimalik täpselt mõõta ümbritseva keskkonna suhtelist õhuniiskust ja õhus oleva veeauru rõhku [22]. Neid parameetereid jälgides, võib pisema avarii korral olla tõsisem süsteemi rike ennetatav.

Inertsiaalanduriga kasutatakse andmevahetuseks USART liidest. MS5803-14BA ja BME230 suhtlevad mikrokontrolleriga üle sünkroonse järjestiksuhtluse liidese (SPI).

Lisaks juba kasutuses olevatele anduritele, on kontrollerplaadil lisaseadmete ühendamiseks veel mitmeid vabasid sisendeid ja väljundeid:

- 3 seadistatava väljundpingega väljundit, 10.8 V (kuni 2 A), 5 V (kuni 1 A) või 3.3 V (kuni 0.5 A);
- 3 PWM lisa väljundit alalisvoolu- või servomootorite ühendamiseks;
- 1x I<sup>2</sup>C, 1x SPI, 1x PDI ja 1x USART liides;
- 8 analoog-digitaalmuunduri sisendit.

## 6. SEADME TARKVARA

Käesolevas peatükis kirjeldatakse allveesõiduki kasutamiseks Raspberry Pi pardaarvutile, disainitud trükkplaadi mikrokontrollerile ning mootori testimiseks kirjutatud programme. Prototüübi tarkvara hõlmab ainult seadme juhtimiseks ja videopildi edastamiseks vajalikke funktsioone.

### 6.1. Mikrokontroller

ATxmega64A3U mikrokontroller on piisavalt võimekas juhtseade, et käitada kõiki allveesõiduki tööks vajalikke toiminguid ning sisend- ja väljundseadmeid. Prototüübi püsivaras on mikrokontroller kuulaja rollis, ehk ainult võtab vastu pardaarvutilt üle USART tulevaid käskude mootorite juhtimiseks. Saabuvate käskude vastuvõtmine on katkestuse põhine ning andmeid talletatakse puhvris.

Käsku oodatakse kujul <M1500150015004500; kus „<“ tähistab käsu algust ja „;“ käsu lõppu. „M“ annab märku, et käsk on mootoritele ja pärast seda järgnevad numbrid sisaldavad kõigi 3 mootori käsku ning lõpus andmete kontrolliks ka nende summat. Juhul kui saabub käsk mingisugusel teisel kujul, siis praagitakse see andmete kontrolli käigus välja. Käsu sobivuse korral edastatakse mootori kiiruse kontrolleritele käsus ettenähtud numbritele vastavad pulsilaiusega PWM signaalid.

Mikrokontroller programmeeriti C-keeles programmeerimiskeskkonnas Atmel Studio. Kontrollerplaadile loodud püsivara on toodud töö lisas D.

### 6.2. Raspberry Pi

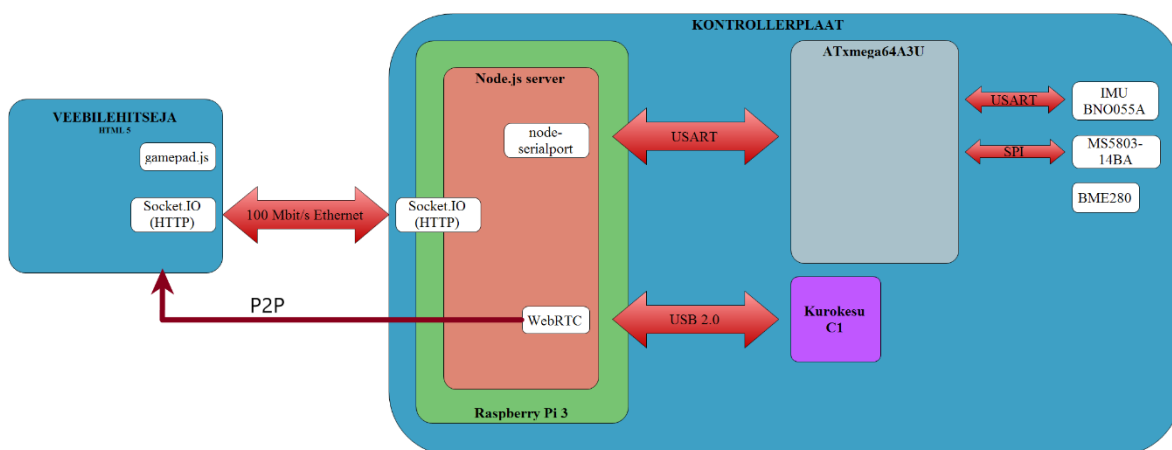
Raspberry Pi 3 on allveesõiduki kõrgem juhtüksus, mille peale on püsitatud Node.js serveri platvorm, mis seadme tööle panemisel koheselt käivitub. Allveesõiduki kasutaja ühendub



serveriga läbi veebilehitseja-põhise kasutajaliidese ning on selle kliendiks. Kasutajaliidese ühendumiseks tuleb minna veebilehitseja aadressile „http://localhost:8080“.

Serveril töötab Socket.IO moodul, mis vahendab andmeid reaalajas läbi kliendi ja serveri vahele loodud andmeedastuskanali [23]. Serveri ja kliendi vaheliseks suhtlusvahendiks on mängupult, mille väljundsignaalile vastavalt seadet juhitakse. Puldil nuppu vajutades kutsutakse esile sündmus, millele reageerib gamepad.js moodul, mis tuvastab millist nuppu vajutati või millist juhtkangi ja kui palju liigutati. Pärast olukorra fikseerimist kutsutakse välja Socket.IO funktsioon andmete saatmiseks serverile. Server töötleb andmed, moodustab käsu ja saadab selle üle USART liidese kontrollerrplaadile. Kaamerast tulevate kaadrite puhul, luuakse veebilehitseja serveri vahel partnervõrk (ingl k *peer-to-peer*, P2P) kasutades WebRTC rakendusliigest. Kuna partnervõrgu püstitamine WebRTC korral nõuab kahe partneri vahel ühenduse sobitajat, implementeeritakse seda Node.js serveri peal. Partnervõrk tagab võimalikult reaalajalise kaadrite edastuse veebilehitsejasse.

Node.js eelis on tema kasutusmugavus, sest antud platvormi puhul pole kasutajal vajadust tarkvara installeerimiseks – piisab ainult arvuti ja veebilehitseja olemasolust. Nii serveri kui ka kliendi poolne programm on kirjutatud JavaScript programmeerimiskeeles tekstiredaktoris Notepad++. Seadmele loodud tarkvara topoloogia on nähtav joonisel 6.1 ja pardaarvuti tarkvara on toodud töö lisas E.



**Joonis 6.1.** Allveesõiduki tarkvara topoloogia.

### **6.3. Kätursüsteemi testprogramm**

Allveesõiduki mootorite voolutarbe ja tõukejõu testimiseks veemahutis kirjutati väike programm Python programmeerimiskeeles. Programm suhtleb üle jadaliidese kontrollerplaadiga ning võimaldab kasutajal saata kontrolleriga ühendatud mootoritele üksikuid käske.

Kätursüsteemi testprogramm on toodud töö lisas F.

## 7. TULEMUSED JA ANALÜÜS

### 7.1. Disainitud allveesõiduk

Töö lõpptulemusena projekteeriti valmis terviklik, detailideni läbimõeldud ja modelleeritud allveesõiduk. Prototüübi lõplik 3D printitud kooste monteerimine ja praktilised katsed jäävad lõputöö järgsesse aega. Femap keskkonnas tehtud analüüsidele tuginedes täidavad kõik projekteeritud detailid töös (va elektroonikakamber) püstitatud nõuded, ehk sobivad kasutamiseks 50 meetri sügavusel vee all. Ka elektroonikakambri vastupidavust on võimalik lihtsasti parandada, kui lisada korpuse põhja ja küljeseina liitekohale mõned plastikust servad, mis takistaksid põhjal sisse poole paindumast.

Arvutuste tulemustest lähtudes on näha, et 0.5 m/s liikumiskiiruse juures ületab kahe horisontaalsihilise mootoriga täisvõimsusel toodetav tõukejõud (27.88 N) allveesõidukile ja kaablile mõjuva takistuse (24.4 N). 1 m/s liikumiskiiruse juures kasvab kaablile mõjuv takistus drastiliselt (88.5 N) ning ületab mootorite võimekuse rohkem kui kolme kordselt. Seda arvestades jääb ekstreemsematel juhtudel 1 m/s liikumiskiiruse saavutamine võimatuks, aga nähtavasti ei tulene see koostatud allveesõiduki disainist vaid puhtalt kaabli liikumisest vees. Siiski tuleb meeles pidada, et antud arvutustes arvestati hüpoteetilise olukorraga kus kaabel on pinges ning liikumissuunaga täiesti risti. Seisva vee korral saab kaablit pikemalt järele anda ning selle pingest vabastada. Normaalingimustel, kus valdav osa kaablist jookseb otse allveesõiduki taga peaks 1 m/s liikumiskiirus siiski saavutatav olema.

Mehaanilise arenduse käigus tehti lugematul arvul tunde modelleerimist Solid Edge CAD ja Femap simulatsioonitarkvaras. Valmis modelleeriti 3 erinevat väliskorpuset ja mehaanilist lahendust. Protsessi käigus tekkis töö autoril läbi katse ja eksituse meetodi selge ettekujutus, kuidas paremini läheneda taoliste terviklahenduse loomisele. Kindlasti kujunes töö kogumaht tunduvalt suuremaks kui esialgu hinnatud.

## **7.2. Disainitud kontrollerrplaat**

Projekti raames valmistatud trükkplaadi disaini võib õnnestunuks lugeda. Plaadi radade vedamisel esines küll üksikuid vigu, kuid need on antud töös esitatud skeemil ära parandatud.

Kogu allveesõiduki elektroonika on laual nõ „kuivalt“ ära katsetatud ning töötab ootuspäraselt. Ka mootorite märg test õnnestus ilma tõrgeteta. Sellegipoolest võiks kontrollerrplaadi mõõtmete vähendamiseks üle minna 4-kihilise trükkplaadile ning seejuures lisada ka täiendavaid lahendusi.

## **7.3. Tarkvara**

Valminud tarkvara on küll piiratud võimalustega, aga täidab kõik esialgu püstitatud nõuded. Töö autoril puudus varasem kokkupuude Node.js platvormi ja JavaScript programmeerimiskeelega. Disainitud süsteemi täielikuks rakendamiseks tuleb JavaScripti juurde õppida.

## **7.4. Võimalikud edasiarendused**

Allveesõiduki prototüübi edasiarendamiseks on mitmeid ideid:

- asendada Raspberry Pi pardaarvuti LattePanda Delta monoplaatarvutiga, mis realiseeriks 1000 Mbit/s Etherneti ja võimaldaks kasutada paremat pardakaamerat;
- lisada kontrollerrplaadile akupaki monitoorimis- ja laadimisahel, et seade oleks otse välisest pistikust laetav;
- teha FOC kiirusekontrolleri;
- LED valgusti käituri disain;
- laserid - objektide suuruse ja vahemaa hindamiseks;
- 4 mootoriga disain;
- juhtmevaba juhtpuldi ja nutiseadme-põhise kasutajaliidese väljatöötamine.

## Kokkuvõte

Seadmed, millega veealust keskkonda visuaalset vaadata või seal tööd teha, olid veel hiljuti väga valdkonnaspetsiifilised ning seeläbi suuremale kasutajaskonnale liiga kallid või kättesaadamatud. Tänapäevaks on antud valdkondadest tulnud tehnoloogilised lahendused koos teadmiste ja põhimõtetega levinud tavakasutajani ning välja on kujunenud omamoodi allveeseadmete liik – väikesed kaugjuhitavad „vaatlusklassi“ allveesõidukid. 3D printerite edukäik ja tänapäevased monoplaatarvutid, kaamerad ning tarkvaralahendused, loovad huvilisele võimaluse ka ise sedalaadi allveesõiduk valmis ehitada.

Käesoleva lõputöö eesmärk oli välja töötada terviklik kaugjuhitav allveesõiduk veealuse keskkonna vaatlemiseks. Seade peab võimaldama sukelduda kuni 50 meetri sügavusele ning seejuures edastama sõiduki kasutajale üle veebilehitseja-põhise kasutajaliidese reaajalist videopilti. Eesmärgi saavutamiseks raalprojekteeriti allvee tingimusteks optimeeritud allveesõiduki korpus, rõhukindlad mahutid ja käitursüsteem. Lisaks koostati sõidukile elektroonikaseadmete süsteem ja disainiti allveesõiduki elektritoitesüsteemi ning andureid ja mootoreid administreeriv kontrollerrüütel. Riistvara kasutamiseks kirjutati kontrollerrüütele elektrooniliste komponentide juhtimiseks mikrokontrolleri püsivara ning allveesõiduki pardaarvutile Node.js platvormil töötav tarkvara.

Lõputöös tehti allveesõidukite valdkonda tutvustav lühiülevaade ning anti põhjalik kirjeldus sedalaadi seadmete ehitamisest, alamsüsteemidest ja komponentide valikust. Töö koostamine nõudis autorilt täiendavate oskuste omandamist trükkplaadi disainis, programmeerimises, CAD modelleerimises ja simulatsioonitarkvaras Femap.

## KASUTATUD KIRJANDUS

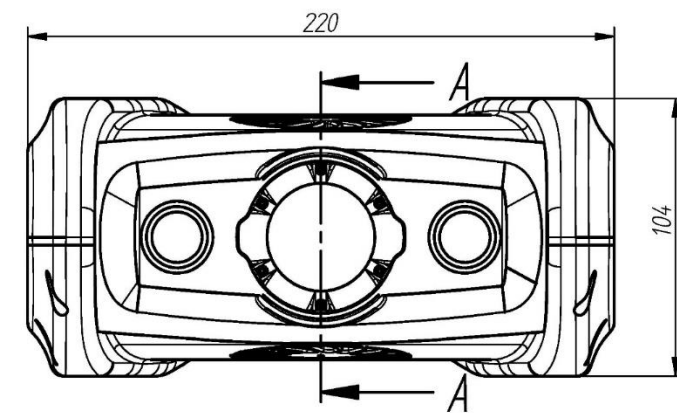
1. Christ, R., Wernli Sr, R. (2014). The ROV Manual A User Guide for Remotely Operated Vehicles. Waltham: Elsevier. 679 lk.
2. Tallinna Tehnikaülikool. (s.a). TTÜ biorobotika keskuse robot U-CAT pälvis laialdase rahvusvahelise meediatähelepanu. [veebileht] <https://www.ttu.ee/ttu-uudised/uudised/arvamus/ttu-biorobotika-keskuse-robot-u-cat-palvis-laialdase-rahvusvahelise-meediatahelepanu/> (16.08.2018)
3. Eesti mereinstituut. (s.a). Remotely Operated Vehicle. [veebileht] <http://www.sea.ee/valibaasid-ja-laevad/rov/> (16.08.2018)
4. Moore, S., Bohm, H., Jensen, V. (2010), Underwater Robotics Science, Design & Fabrication. Monterey: Marine Advanced Technology Education (MATE) Center. 758 lk.
5. Tartu Ülikool. (s.a), Keha tihedus ja üleslükkejõud. [veebileht] [https://sisu.ut.ee/mehaanika/keha-tihedus-ja-üleslükkejõud](https://sisu.ut.ee/mehaanika/keha-tihedus-ja-uleslukkejoud) (16.08.2018)
6. Parker O-Ring Handbook. Parker Hannifin Corporation. [https://www.parker.com/literature/ORD%205700%20Parker\\_O-Ring\\_Handbook.pdf](https://www.parker.com/literature/ORD%205700%20Parker_O-Ring_Handbook.pdf) (16.08.2018)
7. Siemens. (s.a), Finite Element Analysis (FEA). [veebileht] [https://www.plm.automation.siemens.com/en\\_us/plm/fea.shtml](https://www.plm.automation.siemens.com/en_us/plm/fea.shtml) (16.08.2018)
8. Masinaelementide FEM arvutused. TTÜ Mehhatroonikainstituut. [https://moodle.hitsa.ee/pluginfile.php/1735276/mod\\_folder/content/0/SIMULATION%20FEM/T%C3%B5mme/Juhend\\_Labor\\_nr3\\_Masinaelementide%20FEM%20arvutused.doc?forcedownload=1](https://moodle.hitsa.ee/pluginfile.php/1735276/mod_folder/content/0/SIMULATION%20FEM/T%C3%B5mme/Juhend_Labor_nr3_Masinaelementide%20FEM%20arvutused.doc?forcedownload=1) (16.08.2018)
9. Iberisa. (s.a), Femap y NX Nastran. [veebileht] <https://iberisa.wordpress.com/> (16.08.2018)
10. Siemens. (s.a), Post processing questions. [veebileht] <https://community.plm.automation.siemens.com/t5/CAE-Simulation-Femap-Forum/Post-processing-questions/td-p/256956> (16.08.2018)
11. OpenROV. (s.a), Propeller Design. [veebileht] <https://forum.openrov.com/t/propeller-design/1200/2> (16.08.2018)

12. Propeller Pages. (s.a), How a Marine Nozzle Works. [veebileht] [http://www.propellerpages.com/?c=nozzles&f=How\\_Nozzles\\_Work](http://www.propellerpages.com/?c=nozzles&f=How_Nozzles_Work) (16.08.2018)
13. Design of an Integrated Electronic Speed Controller for Compact Robotic Vehicles. The University of Queensland. <http://www.araa.asn.au/acra/acra2011/papers/pap125.pdf> (16.08.2018)
14. Blue Robotics. (s.a). Cable Penetrator for 6mm Cable. [veebileht] <https://www.bluerobotics.com/store/connectors/penetrator-10-25-a/> (16.08.2018)
15. Oceanity. (s.a). Understanding Flat Port and Dome Port Theory. [veebileht] <https://oceanity.com.au/learn/understanding-flat-port-and-dome-port-theory/> (16.08.2018)
16. Kurokesu. (s.a). USB camera C1. [veebileht] <https://www.kurokesu.com/shop/cameras/CAMUSB1> (16.08.2018)
17. Microchip Technology (s.a). ATxmega64A3U. [veebileht] <https://www.microchip.com/wwwproducts/en/ATxmega64A3U#datasheet-toggle> (16.08.2018)
18. TPS54331 3-A, 28-V Input, Step Down DC-DC Converter With Eco-mode. Texas Instruments. <http://www.ti.com/lit/ds/symlink/tps54331.pdf> (16.08.2018)
19. INA199 26-V, Bidirectional, Zero-Drift, Low- or High-Side, Voltage-Output, Current-Shunt Monitor. Texas Instruments. <http://www.ti.com/lit/ds/symlink/ina199.pdf> (16.08.2018)
20. BNO055 Intelligent Absolute Orientation Sensor. Bosch. [https://ae-bst.resource.bosch.com/media/tech/media/datasheets/BST\\_BNO055\\_DS000\\_14.pdf](https://ae-bst.resource.bosch.com/media/tech/media/datasheets/BST_BNO055_DS000_14.pdf) (16.08.2018)
21. MS5803-14BA Miniature 14 bar Module. TE connectivity. [http://www.te.com/commerce/DocumentDelivery/DDEController?Action=showdoc&DocId=Data+Sheet%7FMS5803-14BA%7FB3%7Fpdf%7FEnglish%7FENG\\_DS\\_MS5803-14BA\\_B3.pdf%7FCAT-BLPS0013](http://www.te.com/commerce/DocumentDelivery/DDEController?Action=showdoc&DocId=Data+Sheet%7FMS5803-14BA%7FB3%7Fpdf%7FEnglish%7FENG_DS_MS5803-14BA_B3.pdf%7FCAT-BLPS0013) (16.08.2018)
22. Combined humidity and pressure sensor. Bosch. [https://ae-bst.resource.bosch.com/media/tech/media/datasheets/BST-BME280\\_DS001-12.pdf](https://ae-bst.resource.bosch.com/media/tech/media/datasheets/BST-BME280_DS001-12.pdf) (16.08.2018)
23. Npm. (s.a). socket.io. [veebileht] <https://www.npmjs.com/package/socket.io> (16.08.2018)

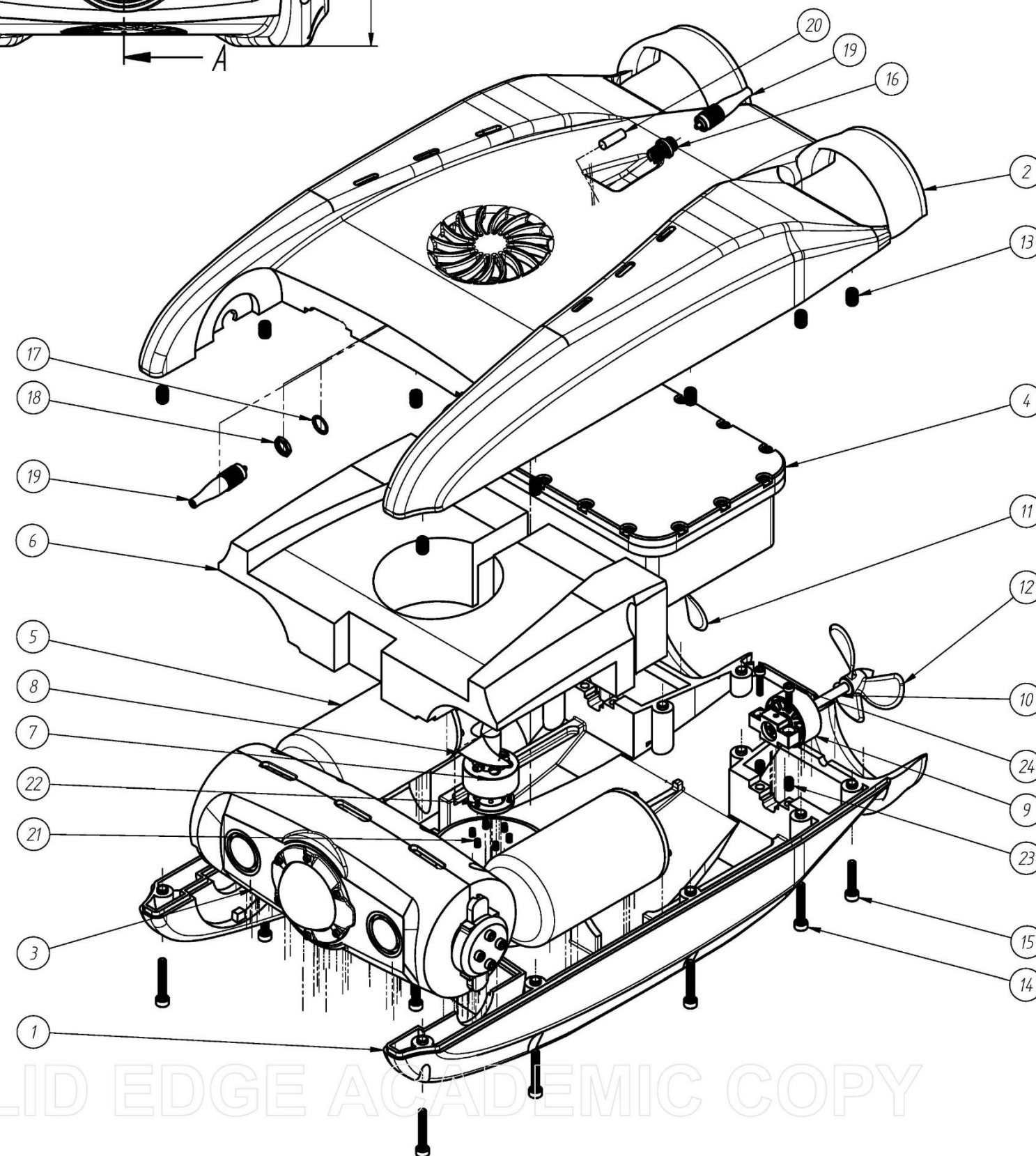
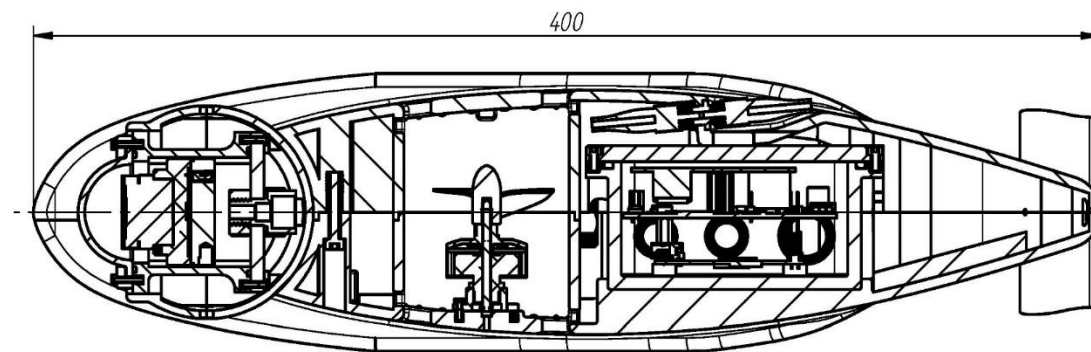
**LISAD**





## **LISA A – Allveesõiduki korpus ja selle detailid**

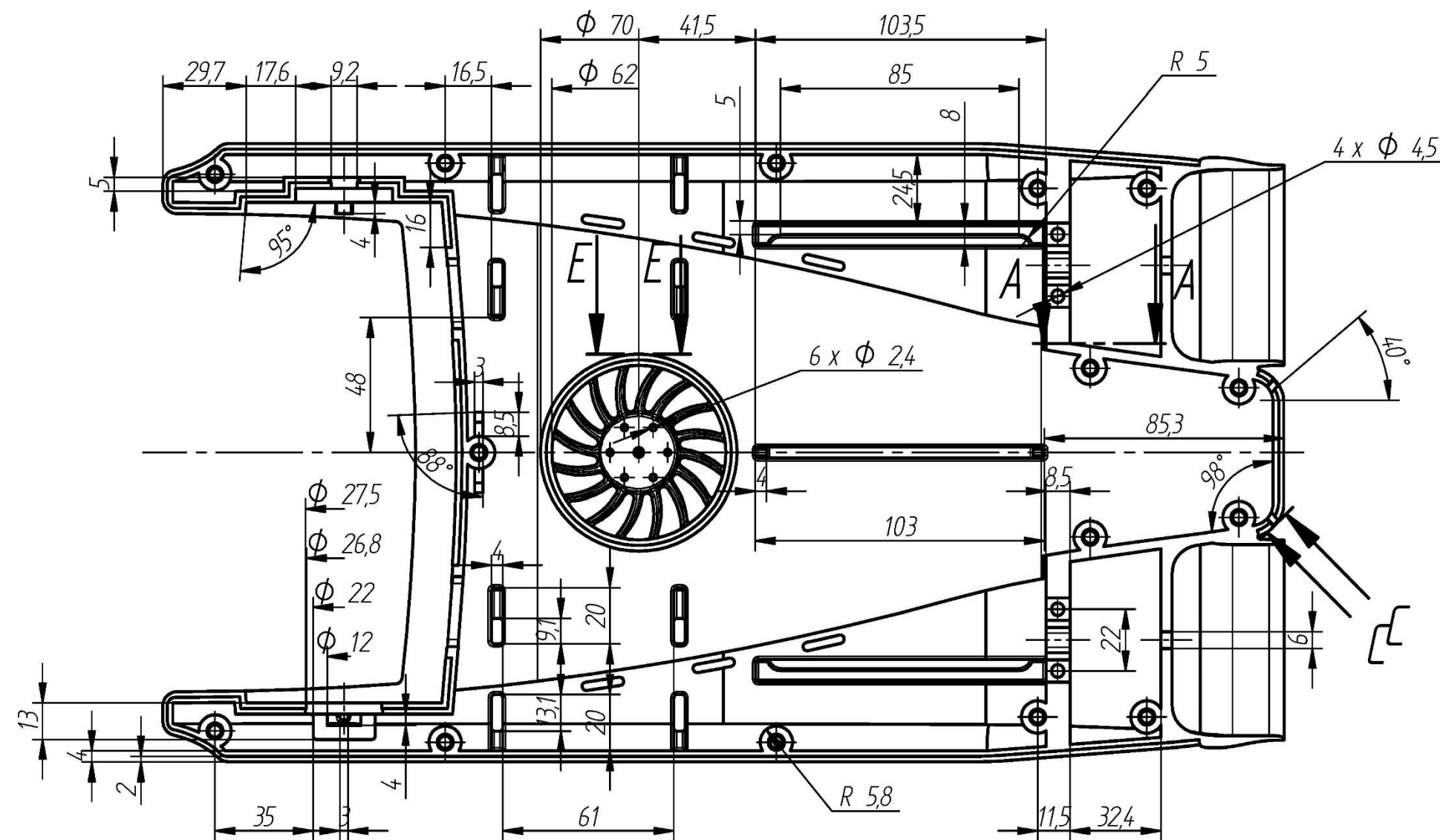


VAADE A-A

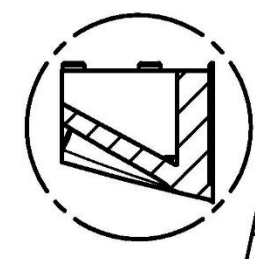


1		Allveesõiduki alumine korpus, HIPS	TN 18/120461 A 01 01 D	1	
2		Allveesõiduki ülemine korpus, HIPS	TN 18/120461 A 01 02 D	1	
3		Roteeruv kamber	TN 18/120461 A 02 00 K	1	
4		Elektroonikakamber	TN 18/120461 A 06 00 K	1	
5		Akupakk	TN 18/120461 A 05 00 K	2	
6		Divinycell H vahuplokk, 100 kg/m3		1	
7		Harjadeta vahelduvvoolumootor Turnigy DST-700		1	
8		2 labaline propeller 50 mm, nailon		1	
9		Harjadeta vahelduvvoolumootor Turnigy DST-700		2	
10		Võll, roostevaba teras		2	
11		3 labaline propeller, parem 65 mm, nailon		1	
12		3 labaline propeller, vasak 65 mm, nailon		1	
13		Tõmbemutter M4x 0.7-8, messing		15	Kuumpressitav
14		Sisekuuskantpolt M4x 0.7-25, roostevaba teras 316		11	
15		Sisekuuskantpolt M4x 0.7-20, roostevaba teras 316		4	
16		FC-FC tüüpi valguskaabli pesa		1	
17		Seib 8x 11x 0.5 mm, tsingitud		1	
18		Mutter M8x 2 mm, tsingitud		1	
19		FC-tüüpi valguskaabli pistik		2	
20		Herkonlüüti		1	
21		Tõmbemutter M2x 0.4-4, messing		6	Kuumpressitav
22		Sisekuuskantpolt M2x 0.4-6, roostevaba teras 316		3	
23		Tõmbemutter M3x 0.5-6, messing		4	Kuumpressitav
24		Sisekuuskantpolt M3x 0.5-14, roostevaba teras 316		4	
Osa	Val	Nimetus, materjal	Tähis	Hulk	Märkus
		Materjal:	Näitamata piirhälbed: ISO 2768	Mass: 3.270	Mööd: 1:2
Teostas	üliõp. Margus Nurme		Allveesõiduk		
Kontrollis	dotsent Veli Palge				
Kinnitas					
 Eesti Maaülikool Estonian University of Life Sciences Tartu Tehnikakolledž			Leht: 1/3	Tähis: TN 18/120461 A 01 00 K	

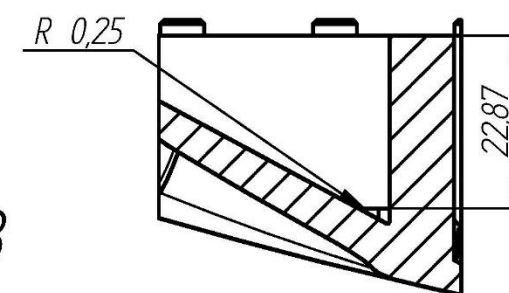




VAADE A-A



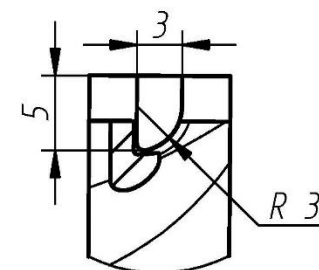
B



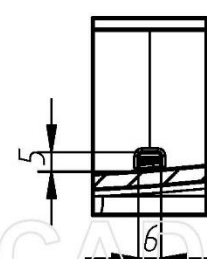
VAADE C-C



D



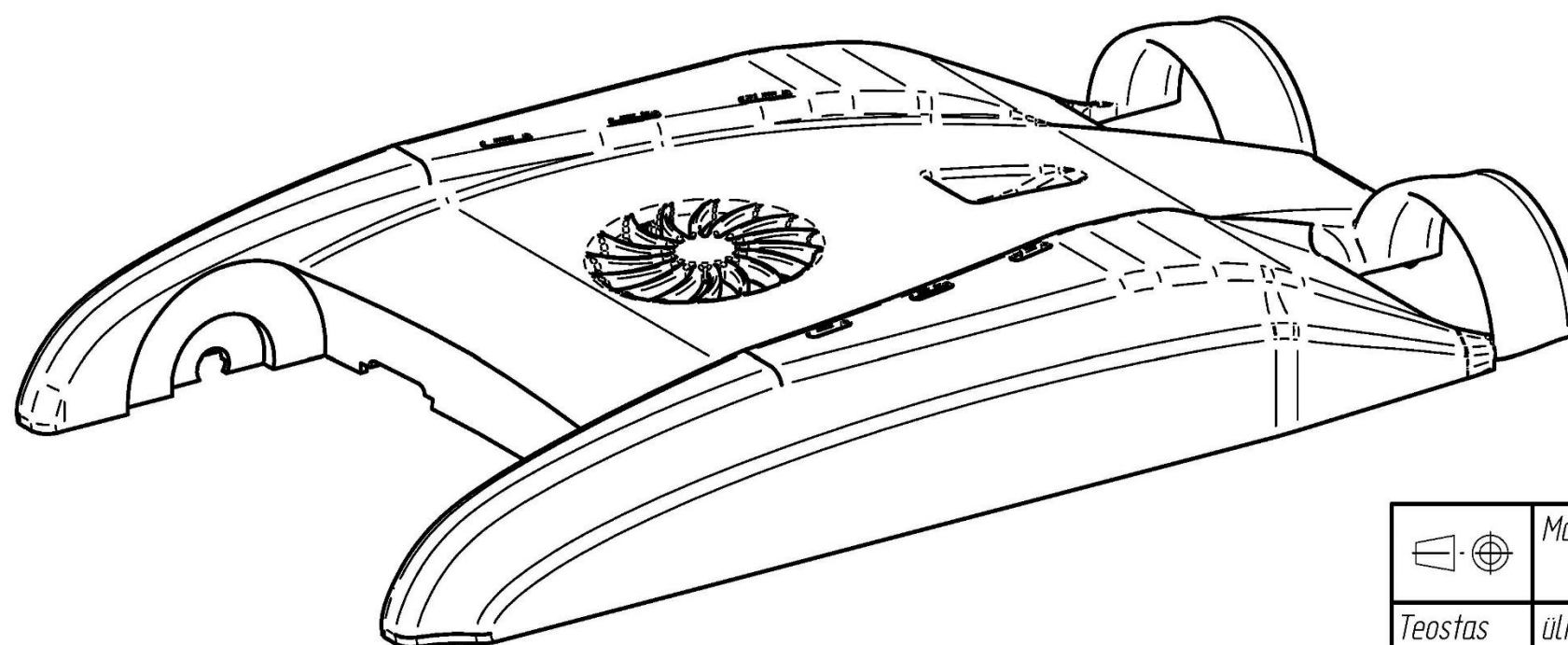
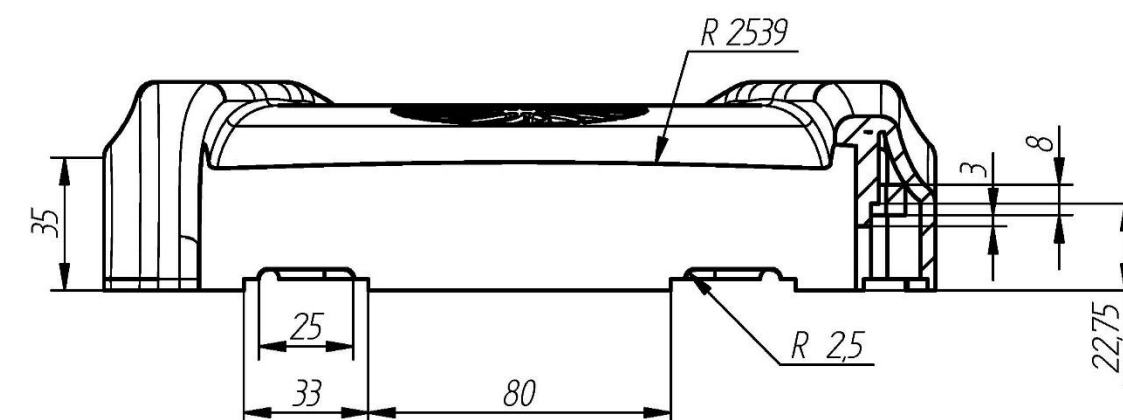
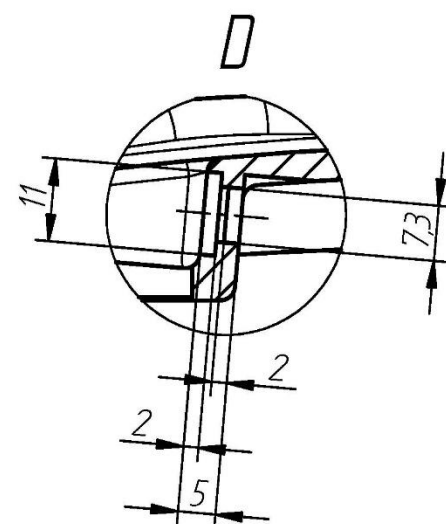
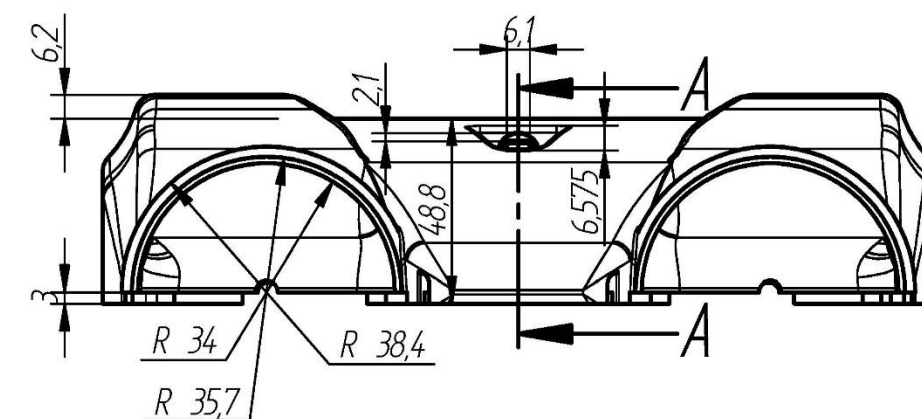
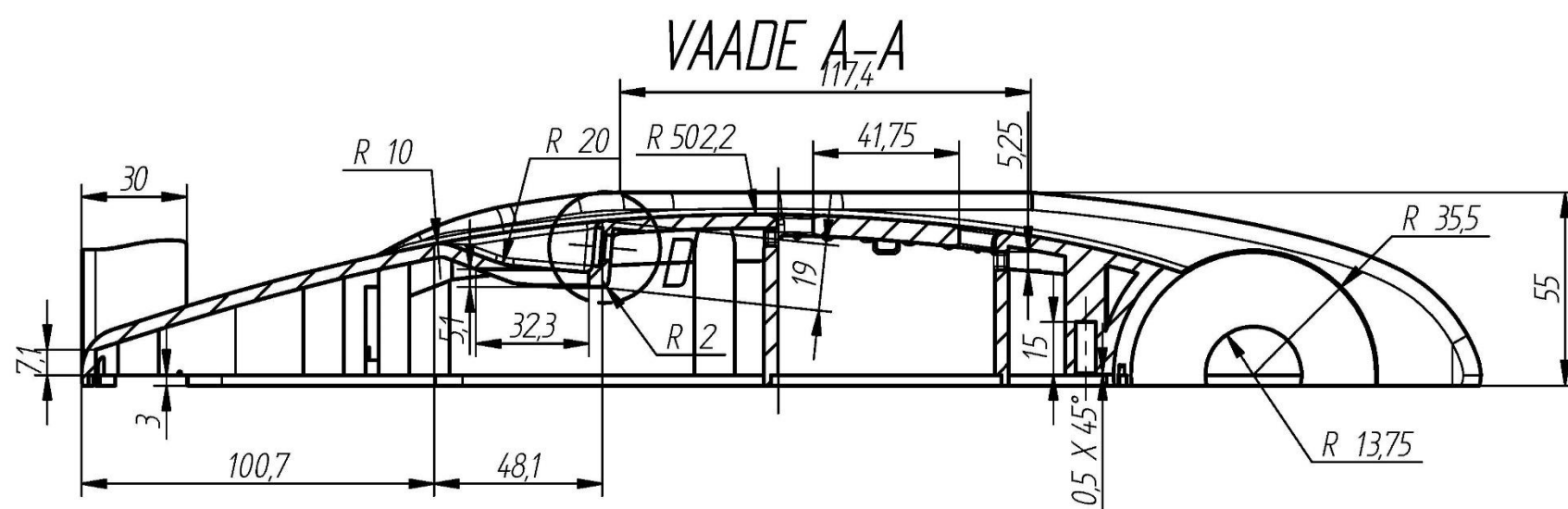
VAADE E-E



1. Märkimata raadiused 1 mm

	Materjal: Formfutura EasyFil HIPS	Näitamata piirhälbed: ISO 2768	Mass: 0.574	Mõõt: 1:2
Teostas	üliõp. Margus Nurme	Allveesõiduki alumine korpus		
Kontrollis	dotsent Veli Palge			
Kinnitas				
 Eesti Maaülikool Estonian University of Life Sciences Tartu Tehnikakolledž Tartu Technology College		Leht: 2/3	Tähis: TN 18/120461 A 01 01 D	

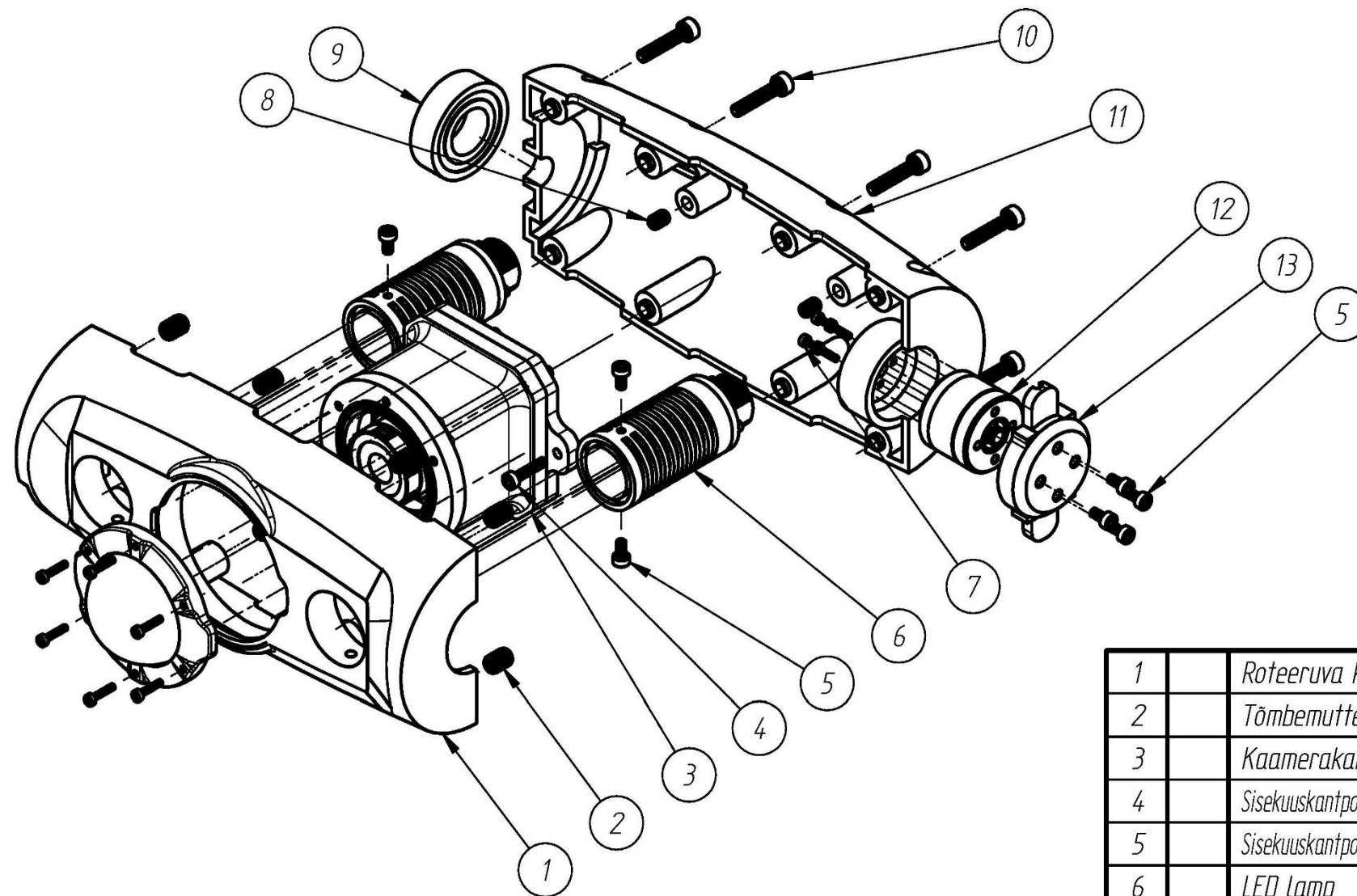
SOLID EDGE ACADEMIC COPY



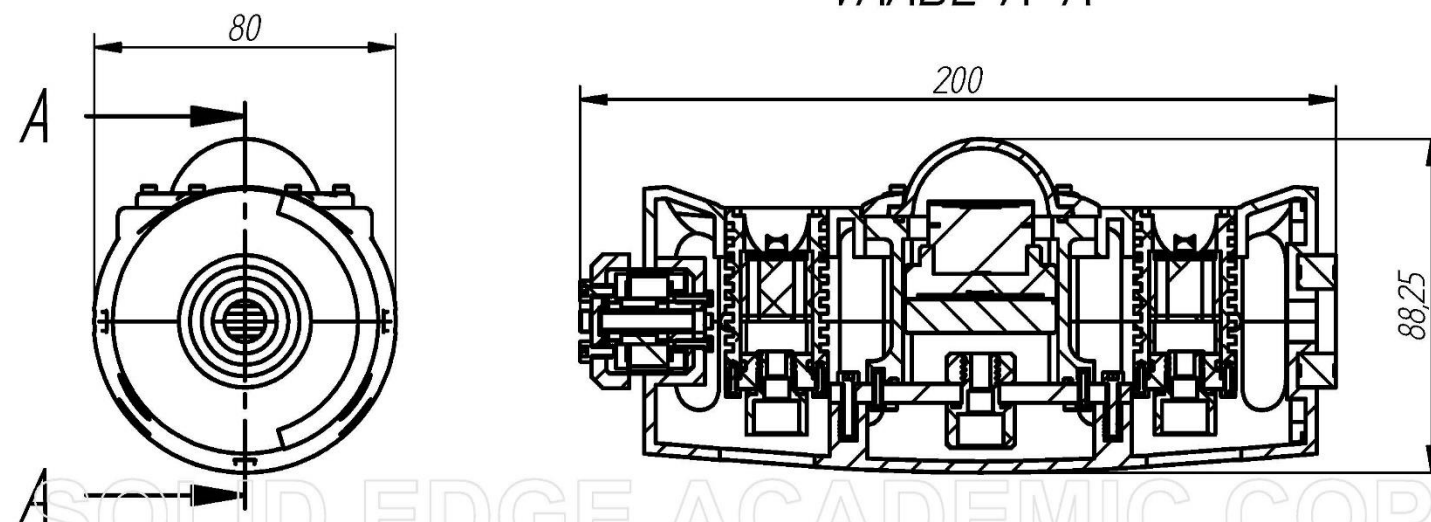
	Materjal: Formfutura EasyFil HIPS	Näitamata piirhálbed: ISO 2768	Mass: 0.556	Mõõt: 1:2
Teostas	üliõp. Margus Nurme	Allveesõiduki ülemine korpus		
Kontrollis	dotsent Veli Palge			
Kinnitas				
Eesti Maaülikool Estonian University of Life Sciences Tartu Tehnikakolledž Tartu Technology College		Leht: 3/3	Tähis: TN 18/120461 A 01 02 D	


SOLID EDGE ACADEMIC COPY

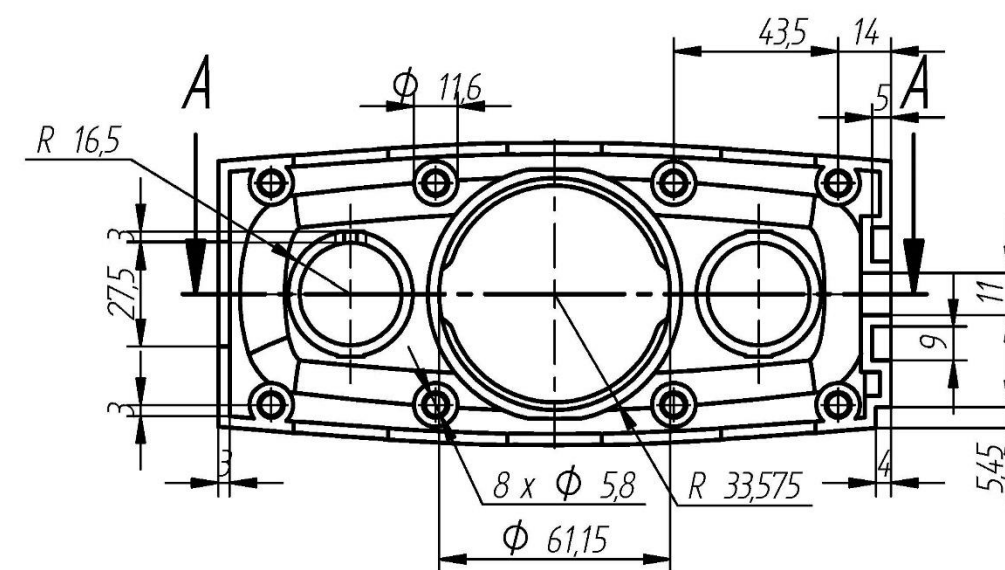
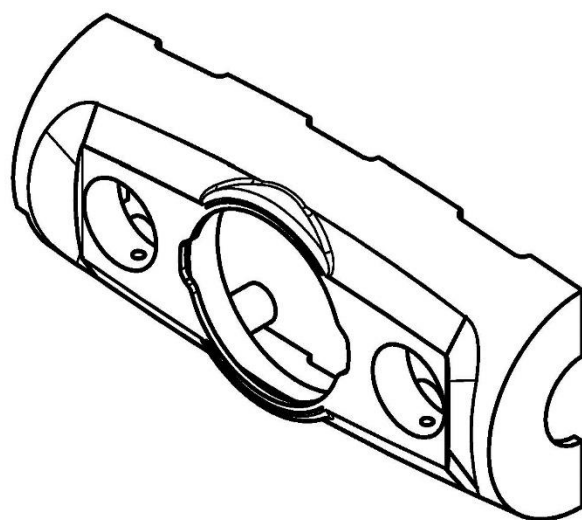
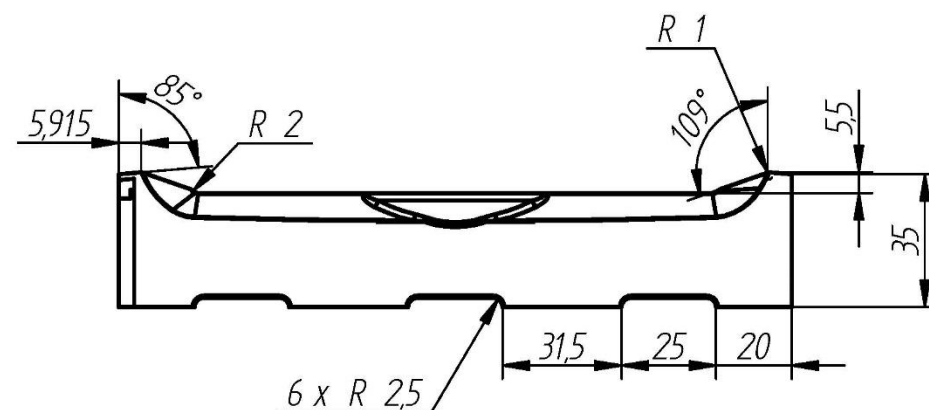




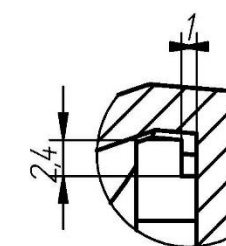
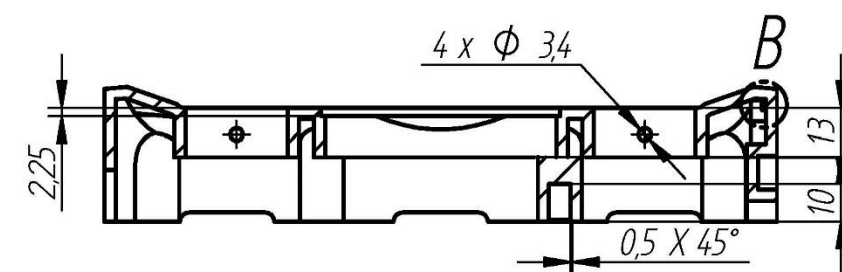
VAADE A-A



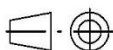

1		Roteeruva kambri esikaas, HIPS	TN 18/120461 A 02 01 D	1	
2		Tõmbemutter M4x 0.7–8, messing		8	Kuumpressitav
3		Kaamerakamber	TN 18/120461 A 03 00 K	1	
4		Sisekuuskantpolt M3x 0.5–14, roostevaba teras 316		2	
5		Sisekuuskantpolt M3x 0.5–6, roostevaba teras 316		8	
6		LED lamp	TN 18/120461 A 04 00 K	2	
7		Sisekuuskantpolt M2x 0.4–8, roostevaba teras 316		4	
8		Tõmbemutter M3x 0.5–6, messing		2	Kuumpressitav
9		Kuullaager 17x 35x 10, POM		1	
10		Sisekuuskantpolt M4x 0.7–20, roostevaba teras 316		8	
11		Roteeruva kambri tagakaas, HIPS	TN 18/120461 A 02 02 D	1	
12		Harjadeta vahelduvvoolumootor		1	
13		Roteeruva kambri mootori kinnitus, HIPS	TN 18/120461 A 02 03 D	1	
Osa	Väl	Nimetus, materjal	Tähis	Hulk	Märkus
		Materjal:	Näitamata piirhälbed: ISO 2768	Mass: 0.590	Mõõt: 1:2
Teostas	üliõp. Margus Nurme		Roteeruv kambri		
Kontrollis	dotsent Veli Palge				
Kinnitas					
 <b>Eesti Maaülikool</b> Estonian University of Life Sciences Tartu Tehnikakolledž Tartu, Eestis			Leht: 1	Tähis: TN 18/120461 A 02 00 K	



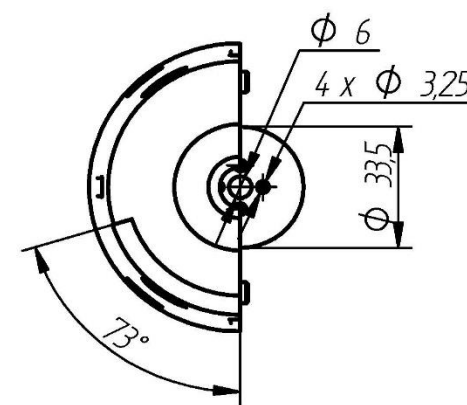
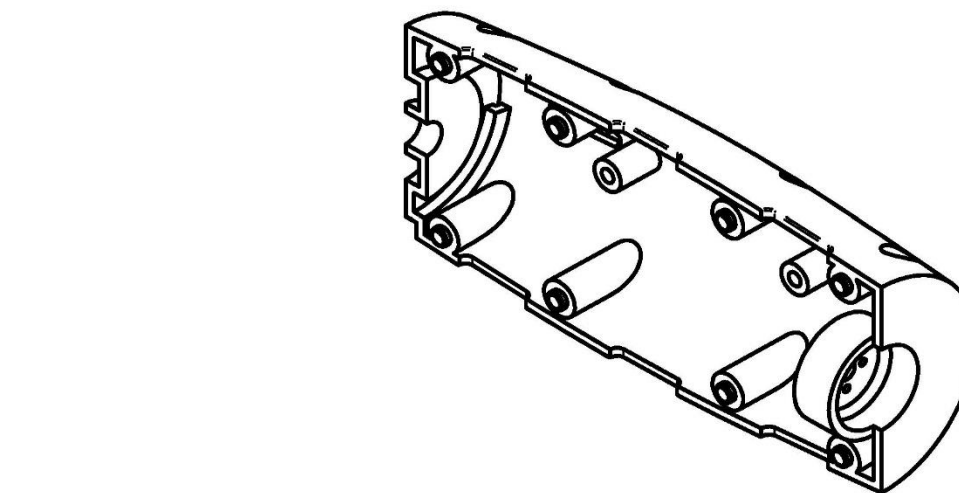
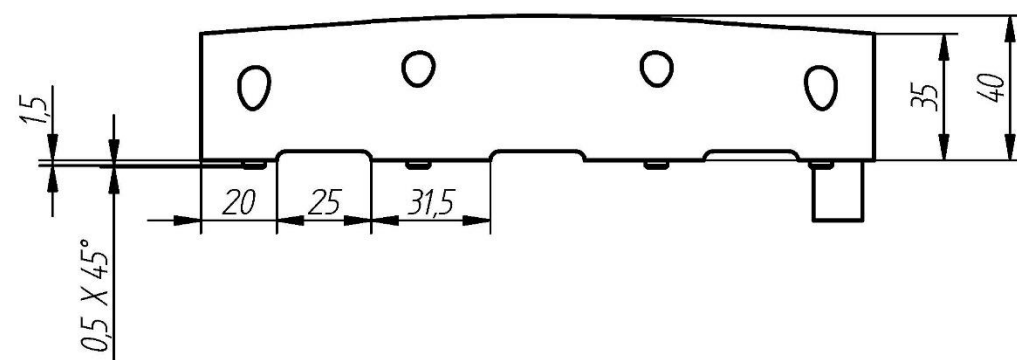
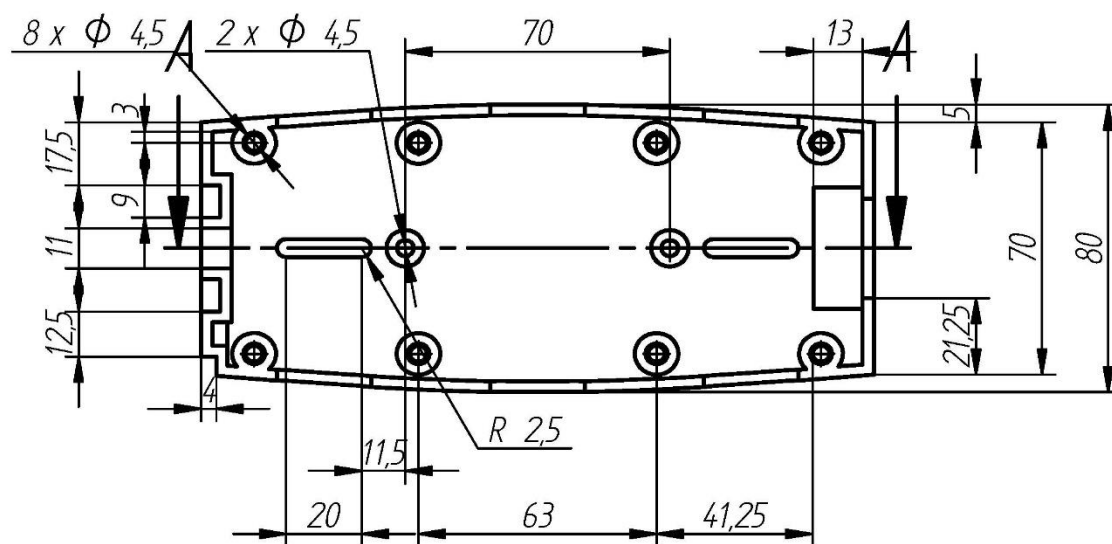
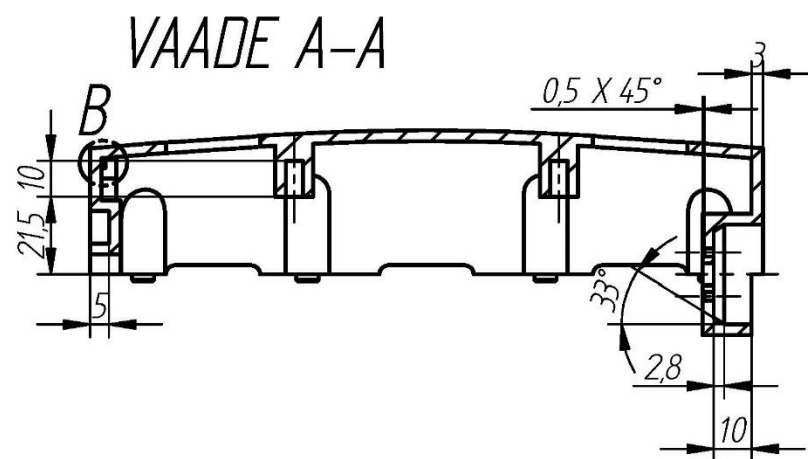
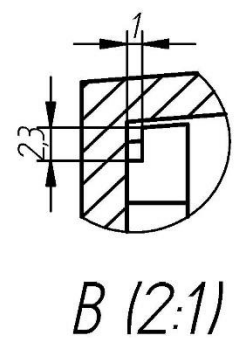
VADE A-A



*B (2:1)*

	Materjal: <i>Formfutura EasyFil HIPS</i>		Näitamata piirhálbed: <i>ISO 2768</i>	Mass: <i>0.087</i>	Mõõt: <i>1:2</i>
	Teostas <i>üliõp. Margus Nurme</i>	Nimetus:  <i>Roteeruva kambri esikaas</i>			
	Kontrollis <i>dotsent Veli Palge</i>				
Kinnitas  					
 <b>Eesti Maaülikool</b> Estonian University of Life Sciences Tartu Tehnikakolledž Tartu Technology College		Leht: <i>1</i>	Tähis: <i>TN 18/120461 A 02 01 D</i>		

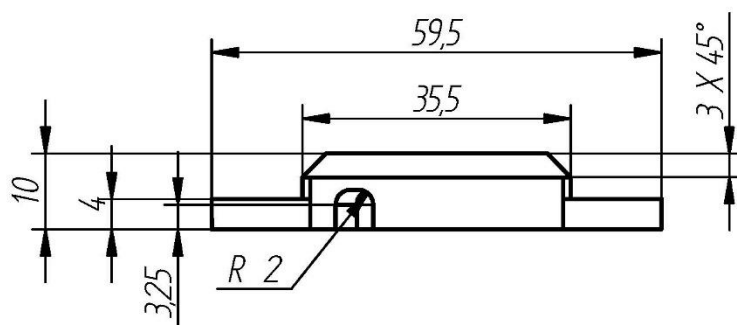
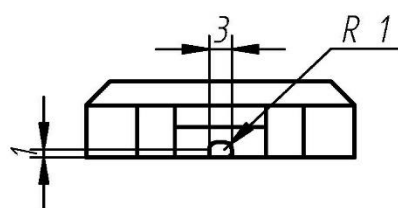
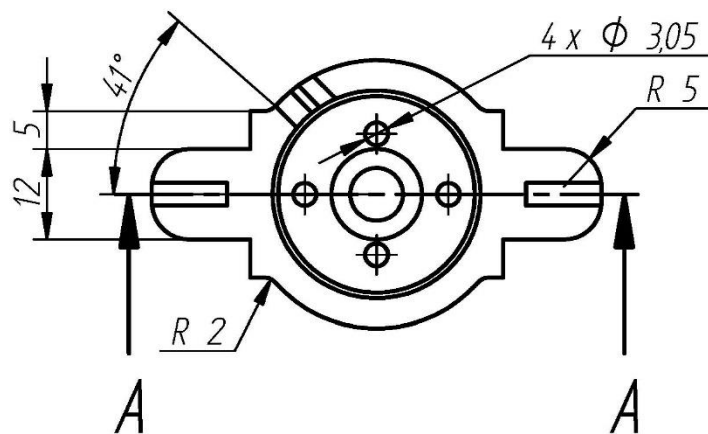
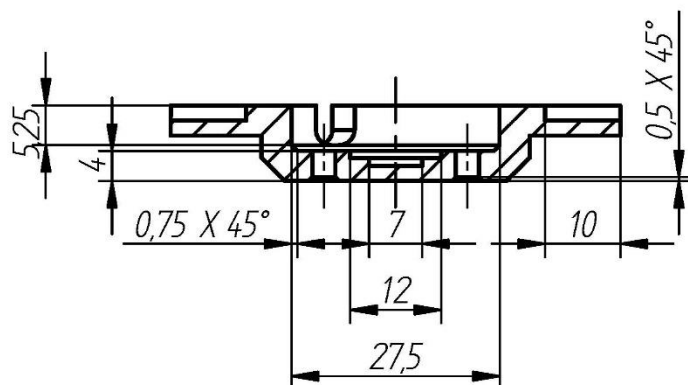




	Materjal: Formfutura EasyFil HIPS	Näitamata piirhálbed: ISO 2768	Mass: 0.094	Mõõt: 1:2
Teostas	üliõp. Margus Nurme	Roteeruva kambri tagakaas		
Kontrollis	dotsent Veli Palge			
Kinnitas				
<b>Eesti Maaülikool</b> Estonian University of Life Sciences Tartu Tehnikakolledž Tartu Technology College		Leht: 1	Tahis: TN 18/120461 A 02 02 D	

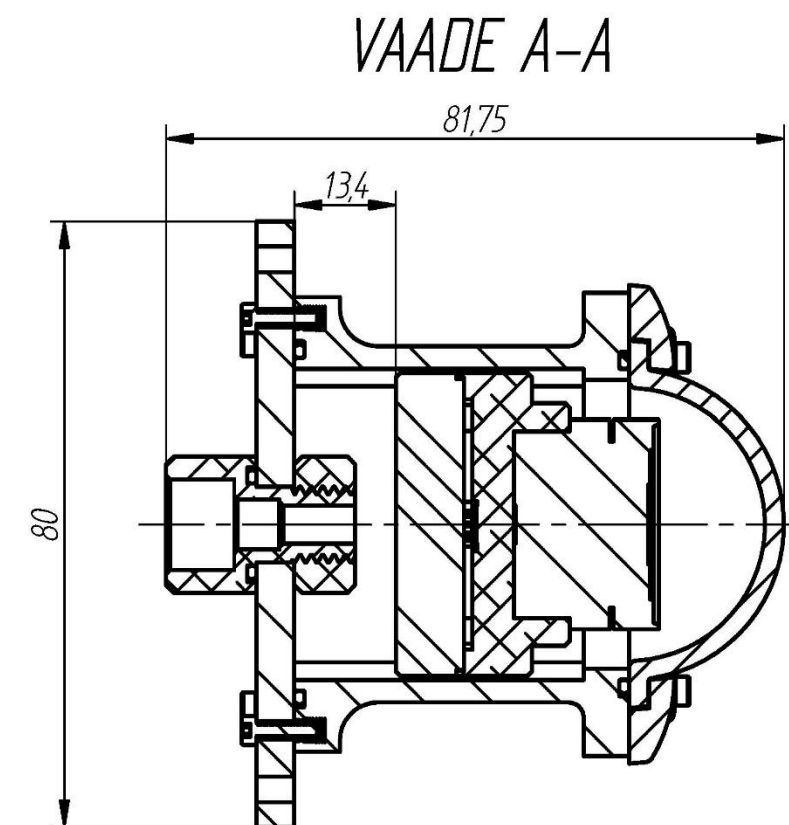
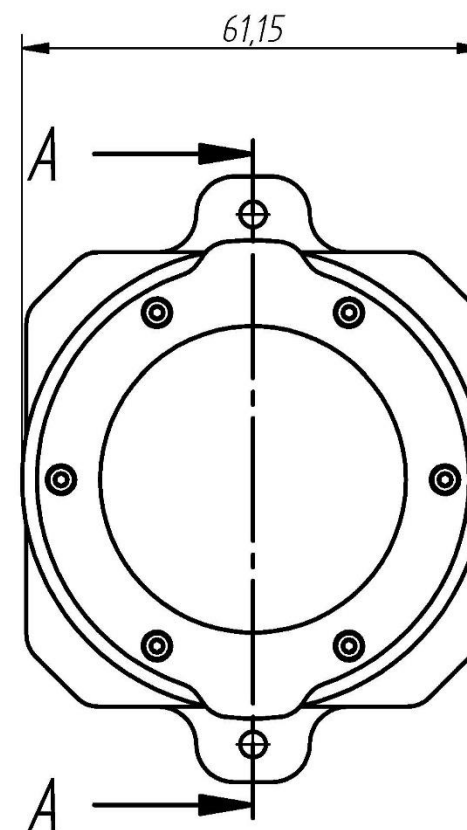
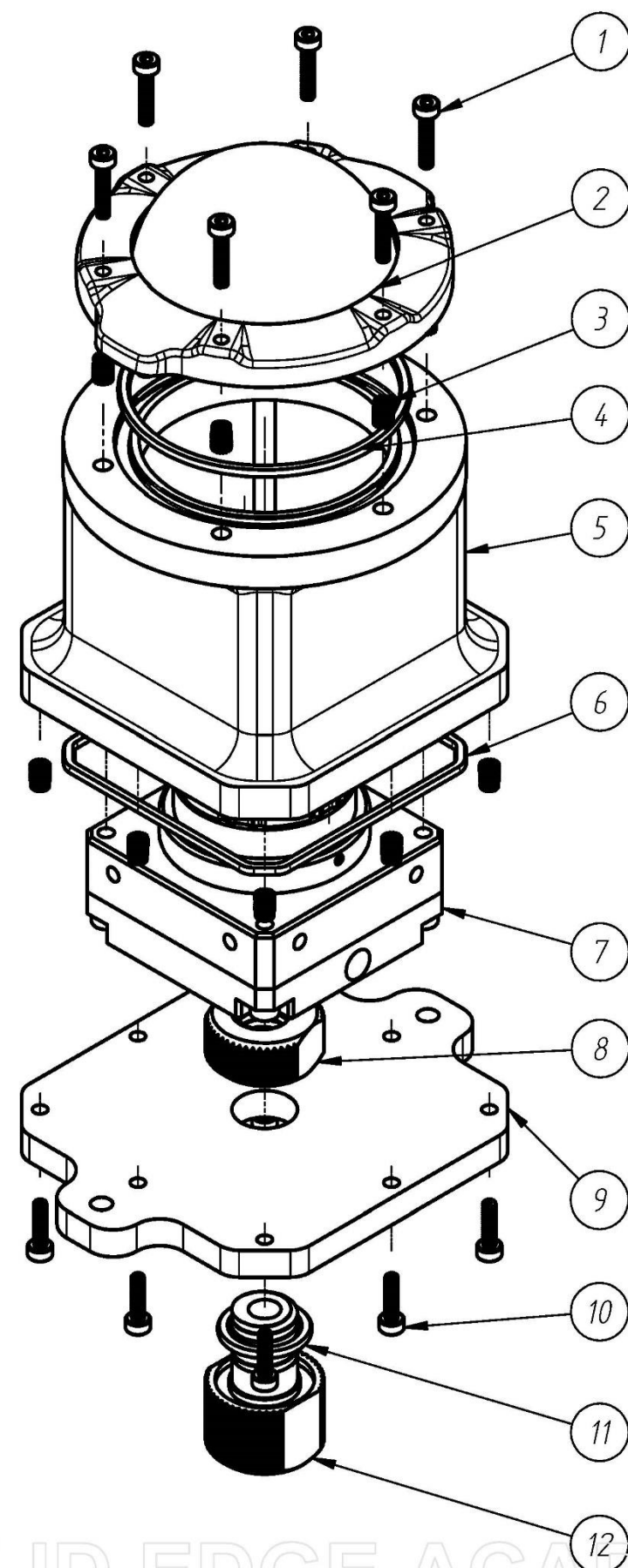
SOLID EDGE ACADEMIC COPY

# VAADE A-A

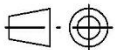



	Materjal: <i>Formfutura EasyFil HIPS</i>	Näitamata piirhálbed: <i>ISO 2768</i>	Mass: <i>0.007</i>	Mõõt: <i>1:1</i>
Teostas	üliõp. Margus Nurme	Roteeruva kambri mootori kinnitus		
Kontrollis	dotsent Veli Palge			
Kinnitas				
	Eesti Maaülikool Estonian University of Life Sciences Tartu Tehnikakolledž Tartu Technology College	Leht: <i>1</i>	Tähis: <i>TN 18/120461 A 02 03 D</i>	



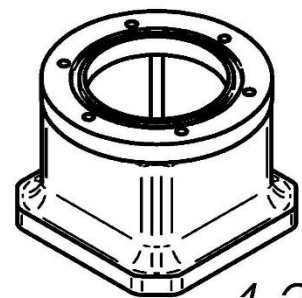
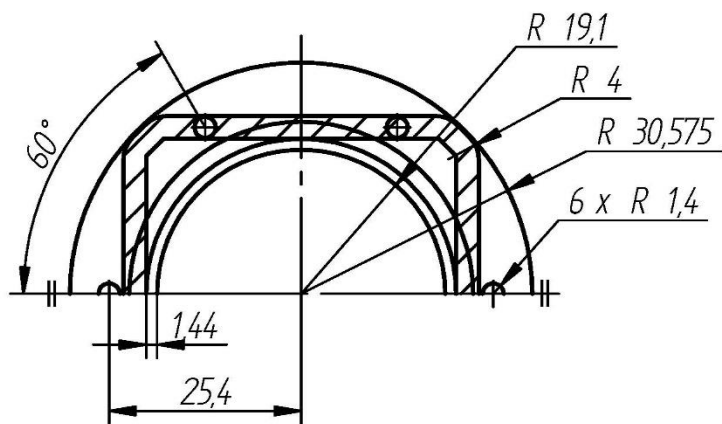


1. Pärast kaabli läbiviimist osa 12 sisse valatud epoksiidliim Loctite EA 3430

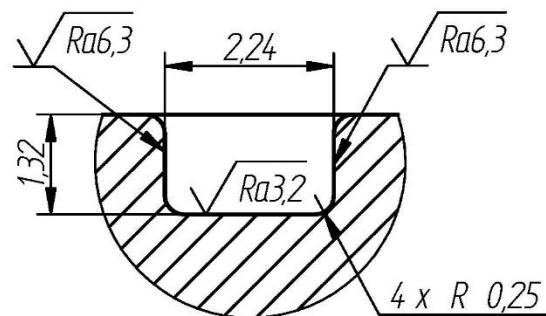
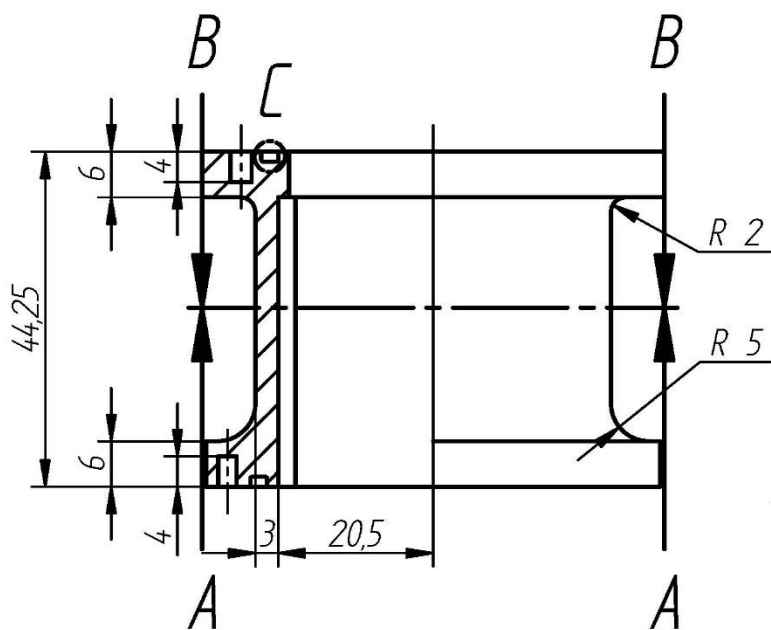
1		Sisekuuskantpolt M2x 0.4-10, roostevaba teras 316		6	
2		Akrüülklaas surverõngaga (ABS)		1	
3		Tõmbemutter M2x 0.4-4, messing		14	Kuumpressitav
4		O-rõngastihend AS568-030 70A, nitrilkumm		1	
5		Kaamerakambri korpus, HIPS	TN 18/120461 A 03 01 D	1	
6		O-rõngas nõör 1.78 mm 70A, nitrilkumm		1	
7		USB kaamera Kurokesu C1		1	
8		Mutter M10x 1.5, alumiinium		1	
9		Kaamerakambri tagakaas, HIPS	TN 18/120461 A 03 02 D	1	
10		Sisekuuskantpolt M2x 0.4-8, roostevaba teras 316		8	
11		O-rõngastihend AS568-013 70A, nitrilkumm		1	
12		Kaabli läbiviik M10x 1.5-12, alumiinium		1	
Osa	Väl	Nimetus, materjal	Tähis	Hulk	Märkus
		Materjal:	Näitamata piirhälbed: ISO 2768	Mass: 0.173	Mõõt: 1:1
Teostas	üliõp. Margus Nurme		Kaamerakamber		
Kontrollis	dotsent Veli Palge				
Kinnitas					
 <b>Eesti Maaülikool</b> Estonian University of Life Sciences Tartu Tehnikakolledž Tartu Technology College			Leht: 1	Tähis: TN 18/120461 A 03 00 K	

SOLID EDGE ACADEMIC COPY

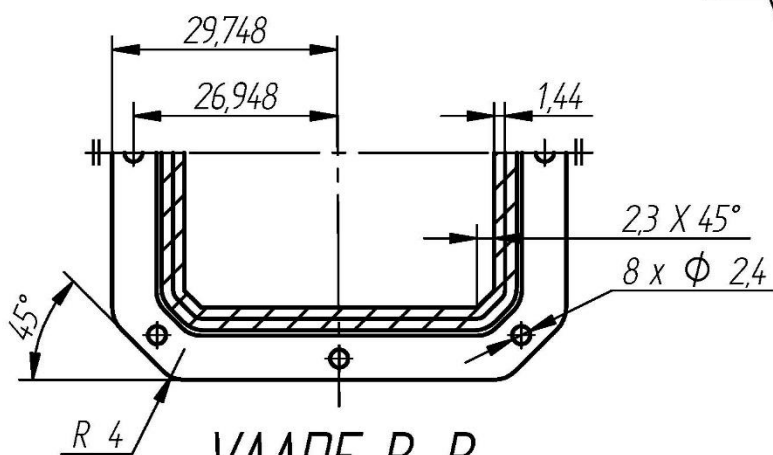
# VAADE A-A



1:2

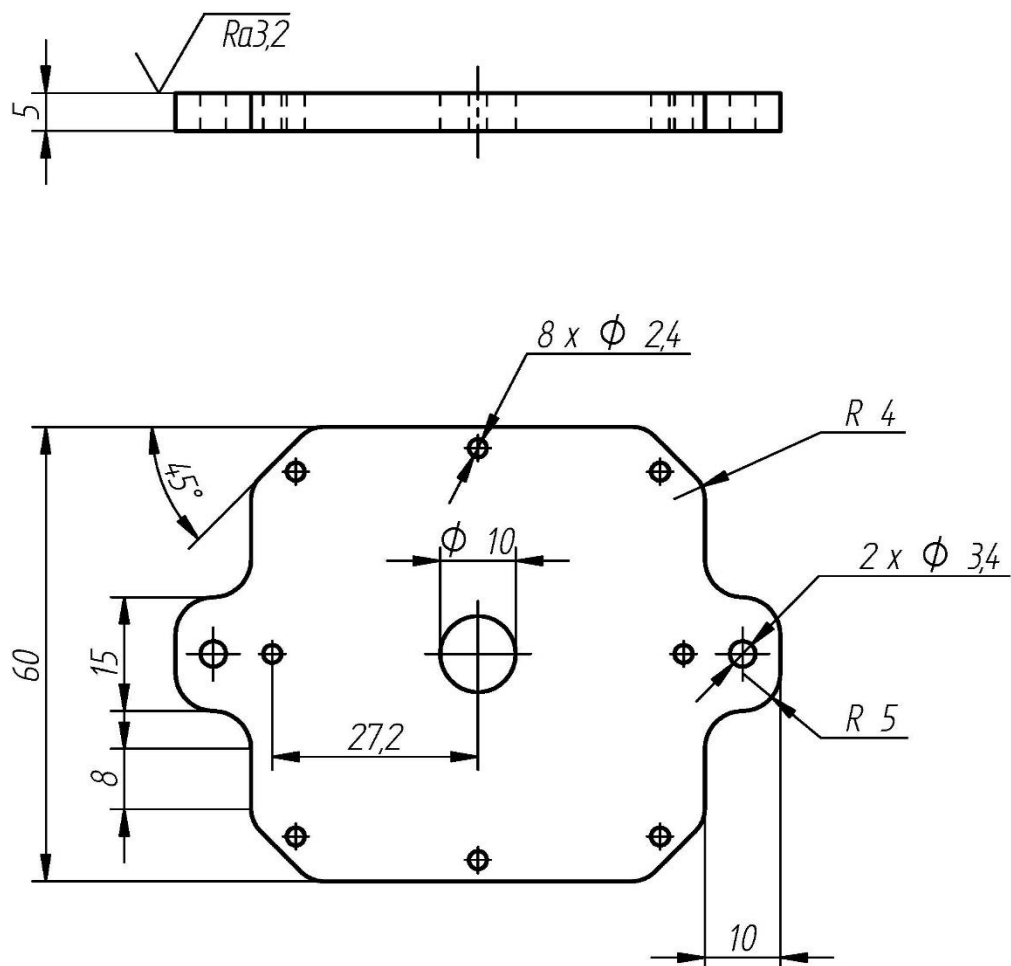


C (10:1)



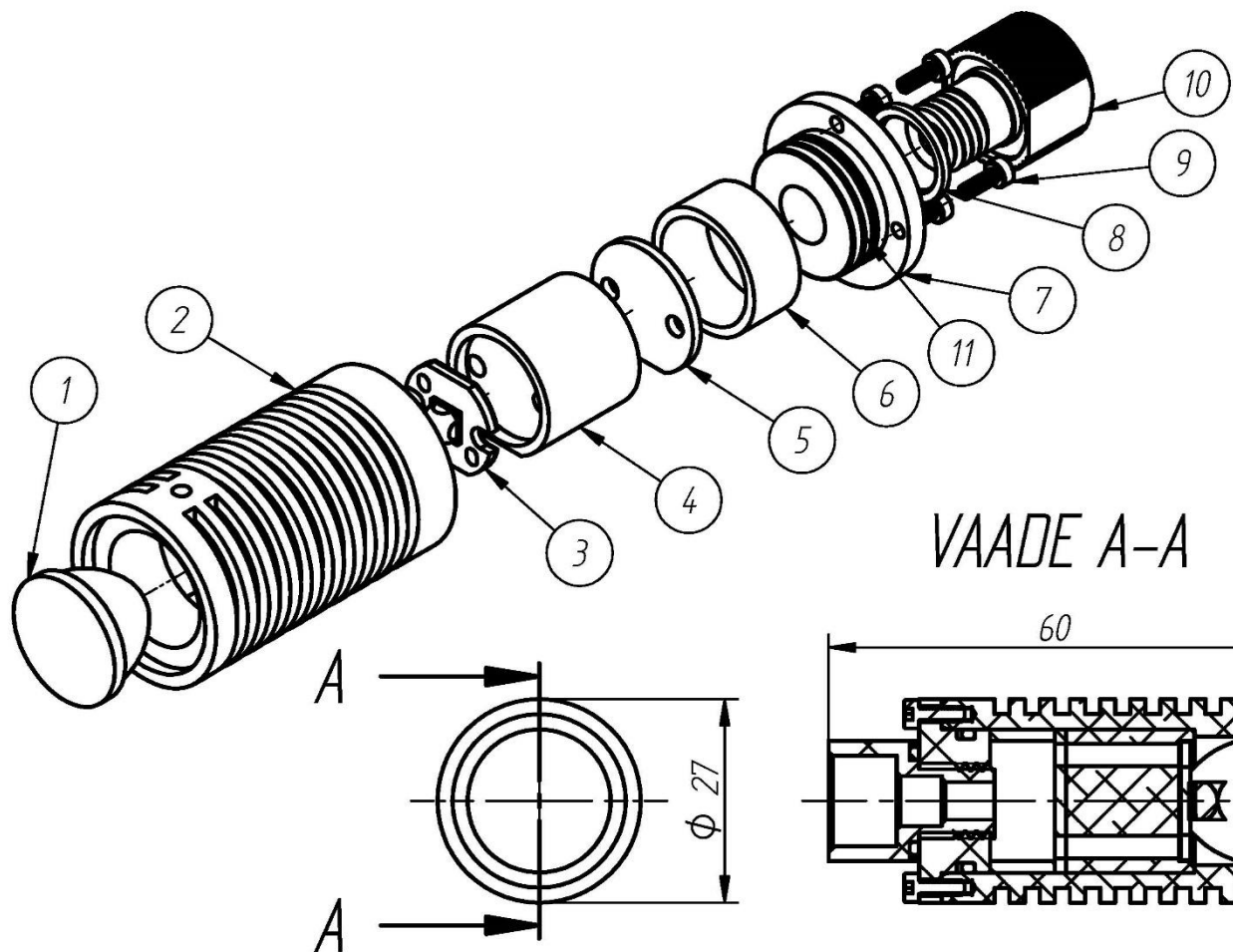
# VAADE B-B

	Materjal: Formfutura EasyFil HIPS	Näitamata piirhälbed: ISO 2768	Mass: 0.039	Mõõt: 1:1
Teostas	üliõp. Margus Nurme	Kaamerakambri korpus		
Kontrollis	dotsent Veli Palge			
Kinnitas				
	Eesti Maaülikool Estonian University of Life Sciences Tartu Tehnikakolledž Tartu Technology College	Leht: 1	Tahis: TN 18/120461 A 03 01 D	

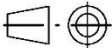



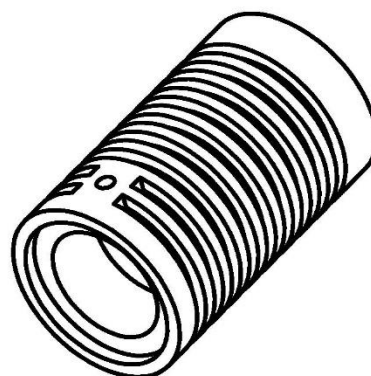
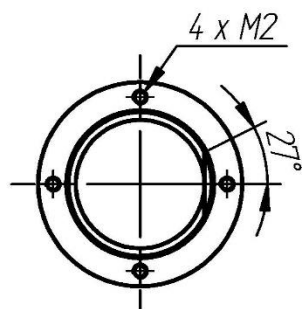
	Materjal: Formfutura EasyFil HIPS		Näitamata piirhálbed: ISO 2768	Mass: 0.019	Mõõt: 1:1
Teostas	üliõp. Margus Nurme	Nimetus: Kaamerakambri tagakaas			
Kontrollis	dotsent Veli Palge				
Kinnitas					
		Leht: 1	Tahis: TN 18/120461 A 03 02 D		



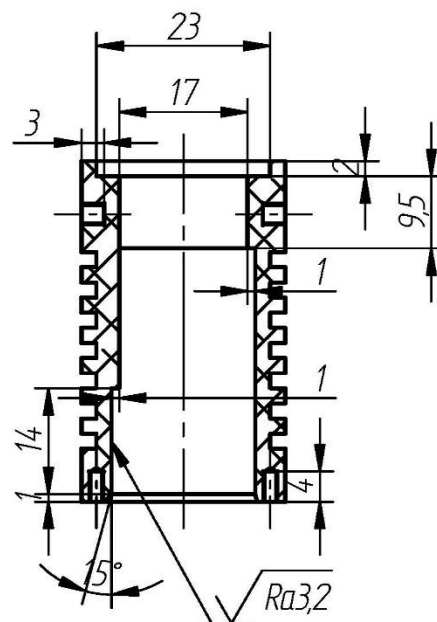
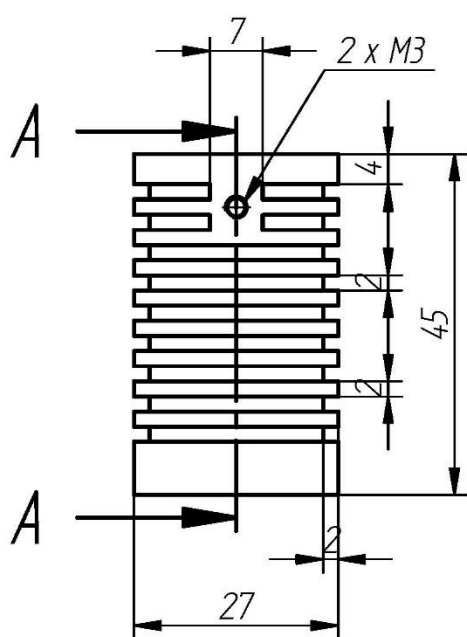


1. Pärast kaabli läbiviimist osa 10 sisse valatud epoksiidliim Loctite EA 3430

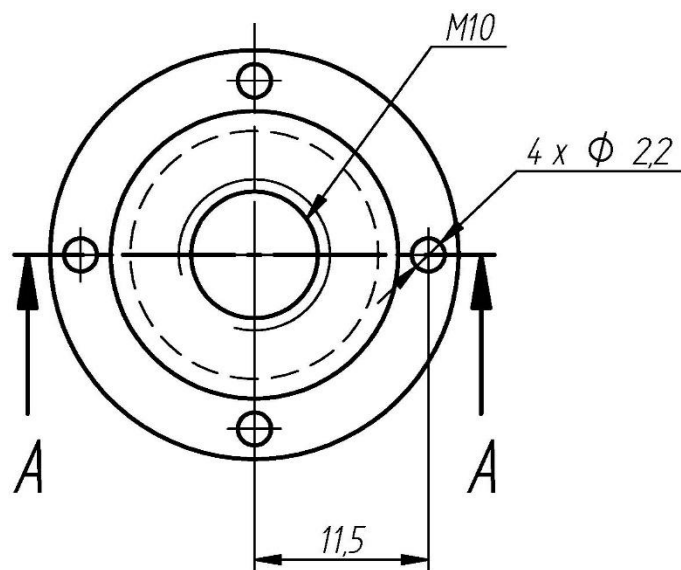
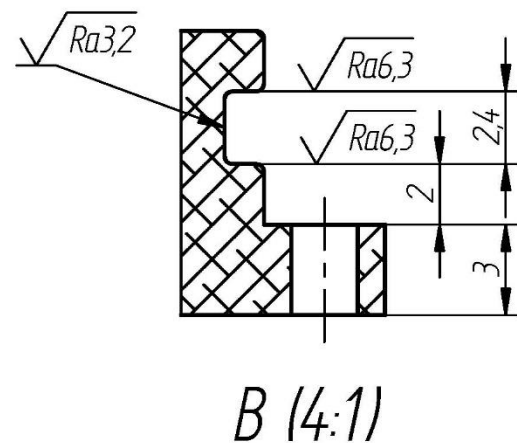
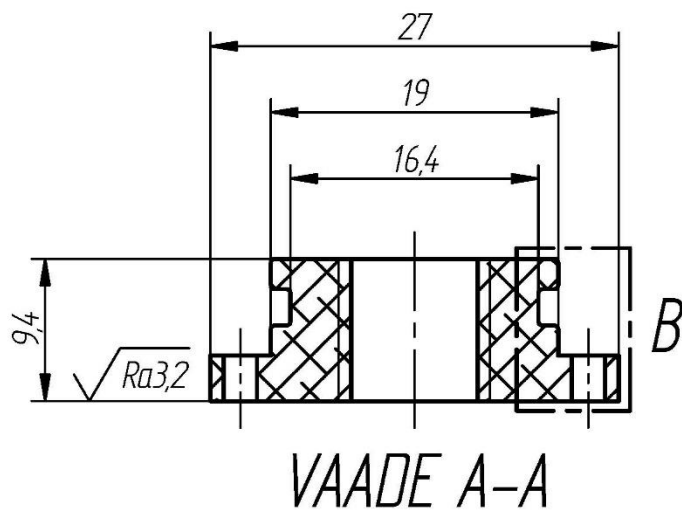
1		Cree XHP50 LED lääts 90 kraadi, akrüül		1	
2		LED lambi korpus, alumiinium	TN 18/120461 A 04 01 D	1	
3		Cree XHP50 LED		1	
4		Radiaator	TN 18/120461 A 04 03 D	1	
5		LED juhtplokk		1	
6		Puks, HIPS		1	
7		LED lambi tagakaas, alumiinium	TN 18/120461 A 04 02 D	1	
8		O-rõngastihend AS568-013 70A, nitrilkumm		1	
9		Sisekuuskantpolt M2x 0.4-8, roostevaba teras 316		4	
10		Kaabli labiviik M10x 1.5-10, alumiinium		1	
11		O-rõngastihend 15.60x 1.78 mm 70A, nitrilkumm		1	
Osa	Väl	Nimetus, materjal	Tähis	Hulk	Märkus
		Materjal:	Näitamata piirhälbed: ISO 2768	Mass: 0.068	Mõõt: 1:1
Teostas	üliõp. Margus Nurme		LED lamp		
Kontrollis	dotsent Veli Palge				
Kinnitas					
 Eesti Maaülikool Estonian University of Life Sciences Tartu Tehnikakolledž Tartu Technology College			Leht: 1	Tähis: TN 18/120461 A 04 00 K	



### VAADE A-A

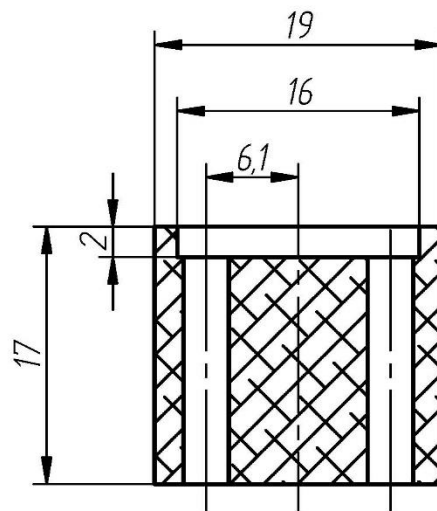


	Materjal: <b>Alumiinium</b>	Näitamata piirhálbed: <b>ISO 2768</b>	Mass: <b>0.029</b>	Mööõt: <b>1:1</b>
Teostas	üliõp. Margus Nurme	Nimetus: <b>LED lambi korpus</b>		
Kontrollis	dotsent Veli Palge			
Kinnitas				
	<b>Eesti Maaülikool</b> Estonian University of Life Sciences Tartu Tehnikakolledž Tartu Technology College	Leht: <b>1</b>	Tahis: <b>TN 18/120461 A 04 01 D</b>	

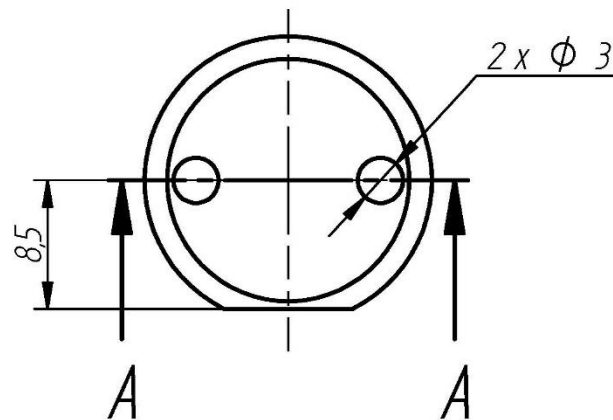


1. Märkimata raadiused 0,25 mm

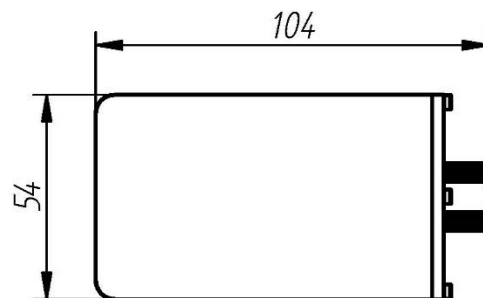
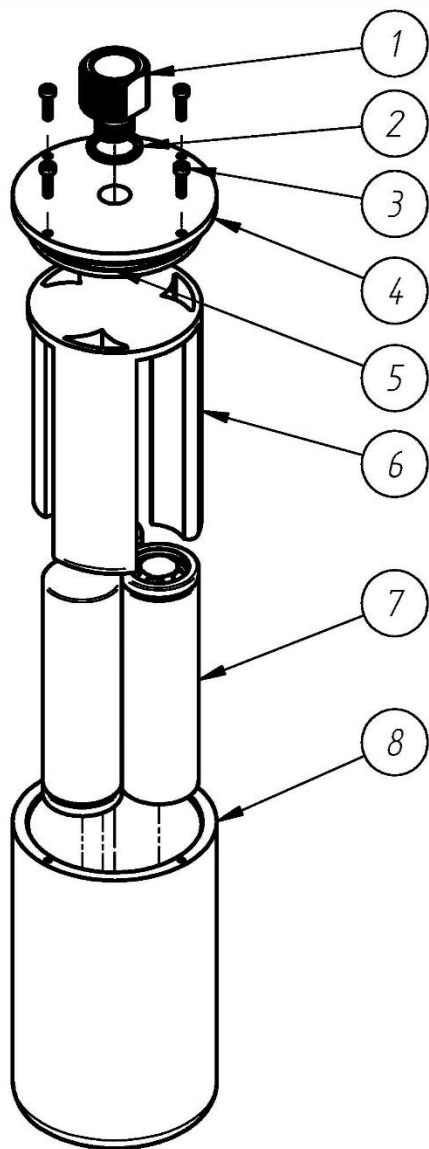
	Materjal: Alumiinium	Näitamata piirhälbed: ISO 2768	Mass: 0.008	Mööd: 2:1
Teostas	üliõp. Margus Nurme	LED lambi tagakaas		
Kontrollis	dotsent Veli Palge			
Kinnitas				
	Eesti Maaülikool Estonian University of Life Sciences Tartu Tehnikakolledž Tartu Technology College	Leht: 1	Tahis: TN 18/120461 A 04 02 D	



VAADE A-A



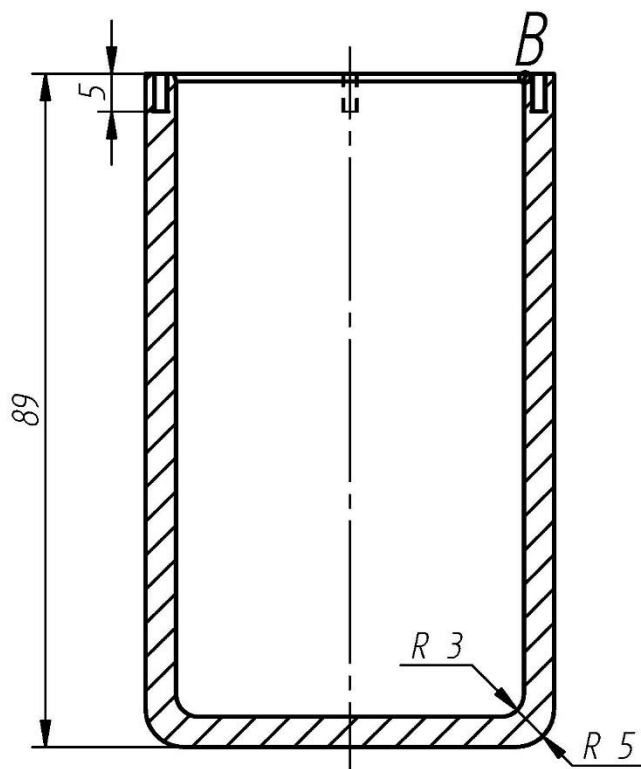
	Materjal: <b>Alumiinium</b>		Näitamata piirhálbed: <b>ISO 2768</b>	Mass: <b>0,011</b>	Mööd: <b>2:1</b>
Teostas	üliõp. Margus Nurme	Nimetus: <b>LED'i radiaator</b>			
Kontrollis	dotsent Veli Palge				
Kinnitas					
<b>Eesti Maaülikool</b> Estonian University of Life Sciences Tartu Tehnikakolledž Tartu Technology College		Leht: <b>1</b>	Tahis: <b>TN 18/120461 A 04 03 D</b>		



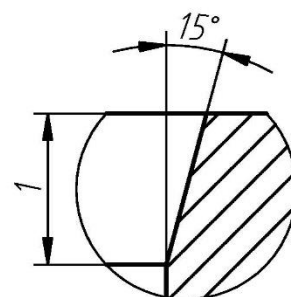
1. Pärast kaabli läbiviimist osa 10 sisse valatud epoksiidliim Loctite EA 3430

1		Kaabli läbiviik M10x 1.5-10, alumiinium		1	
2		O-rõngastihend AS568-013 70A, nitrilkumm		1	
3		Sisekuuskantpolt M2x 0.4-8, roostevaba teras 316		4	
4		Akupaki korpuse kaas	TN 18/120461 A 05 02 D	1	
5		O-rõngastihend 40.94x 2.62 mm 70A, nitrilkumm		1	
6		Akude pesa, HIPS	TN 18/120461 A 05 03 D	1	
7		Aku Vapcell INR2170 4000 mA 15A		3	
8		Akupaki korpus, HIPS	TN 18/120461 A 05 01 D	1	
Osa	Väl	Nimetus, materjal	Tähis	Hulk	Märkus
		Materjal:	Näitamata piirhälbed: ISO 2768	Mass: 0.329	Mööd: 1:2
Teostas	üliõp. Margus Nurme		Akupakk		
Kontrollis	dotsent Veli Palge				
Kinnitas					
 Eesti Maaülikool Estonian University of Life Sciences Tartu Tehnikakolledž Tartu Technology College			Leht: 1	Tähis: TN 18/120461 A 05 00 K	

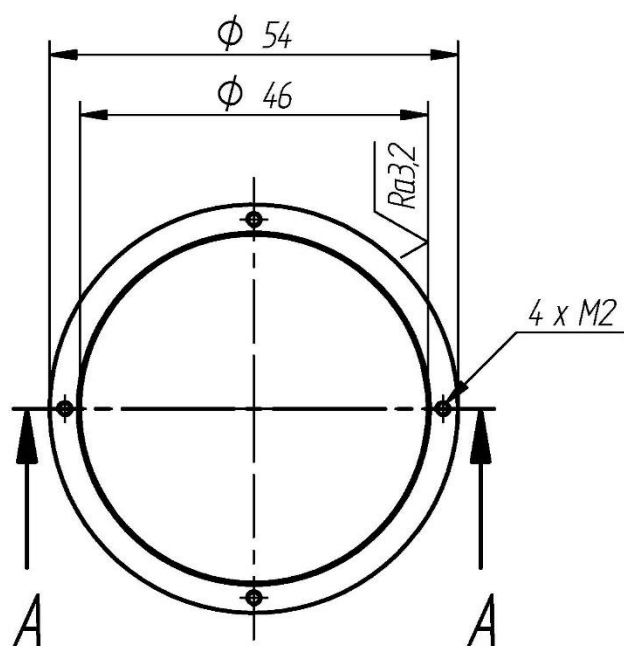




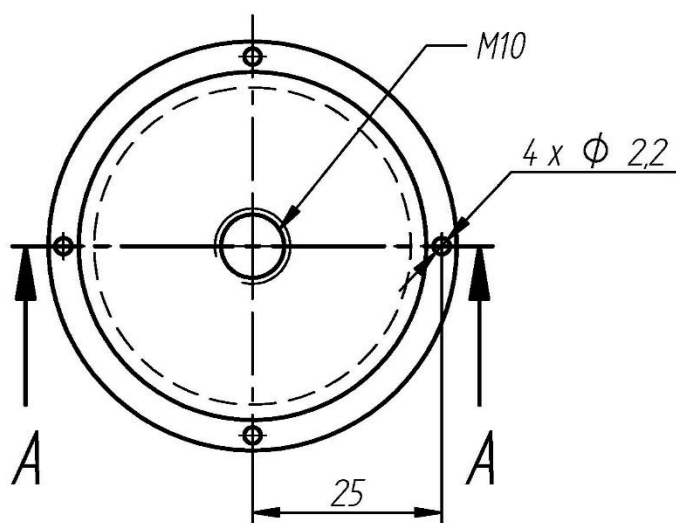
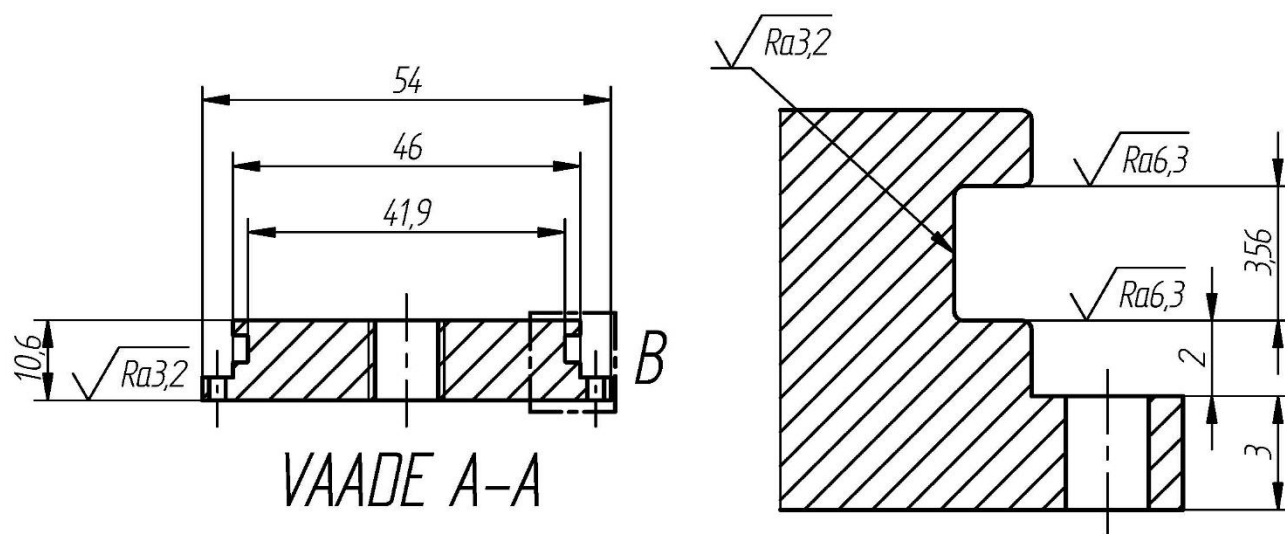
VAADE A-A



B (20:1)

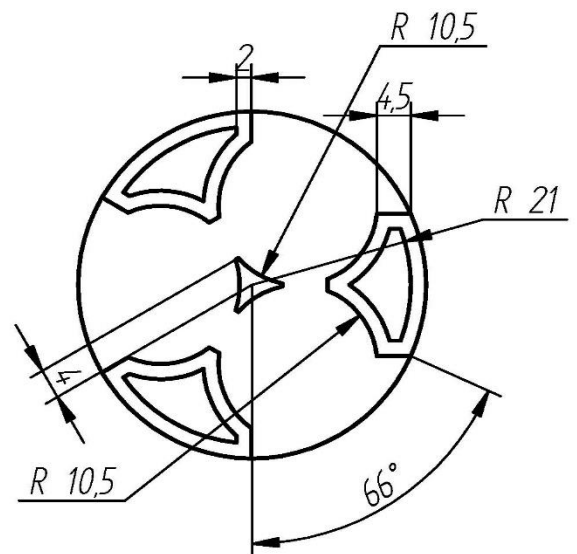
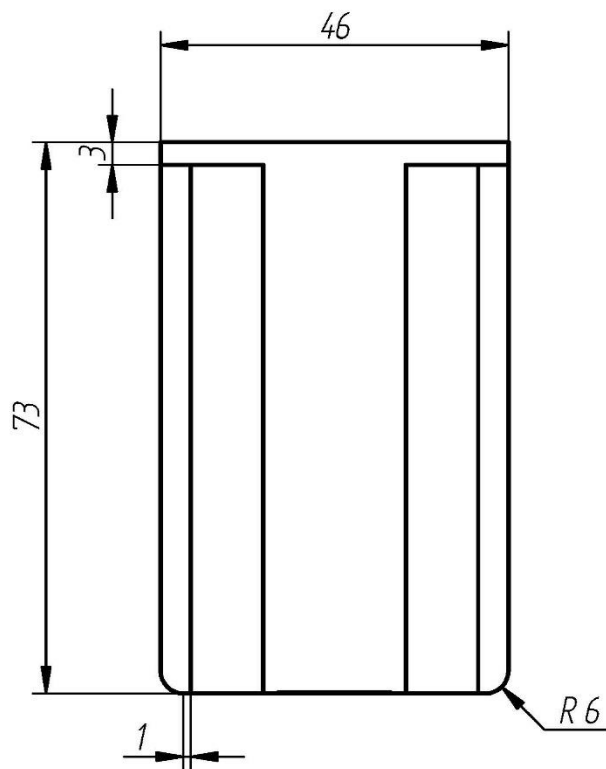
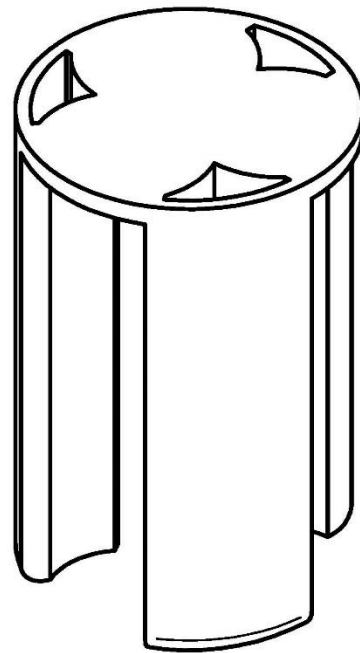


	Materjal: Formfutura EasyFil HIPS	Näitamata piirhálbed: ISO 2768	Mass: 0.064	Mõõt: 1:1
Teostas	üliõp. Margus Nurme	Akupaki korpus		
Kontrollis	dotsent Veli Palge			
Kinnitas				
		Leht: 1	Tahis: TN 18/120461 A 05 01 D	



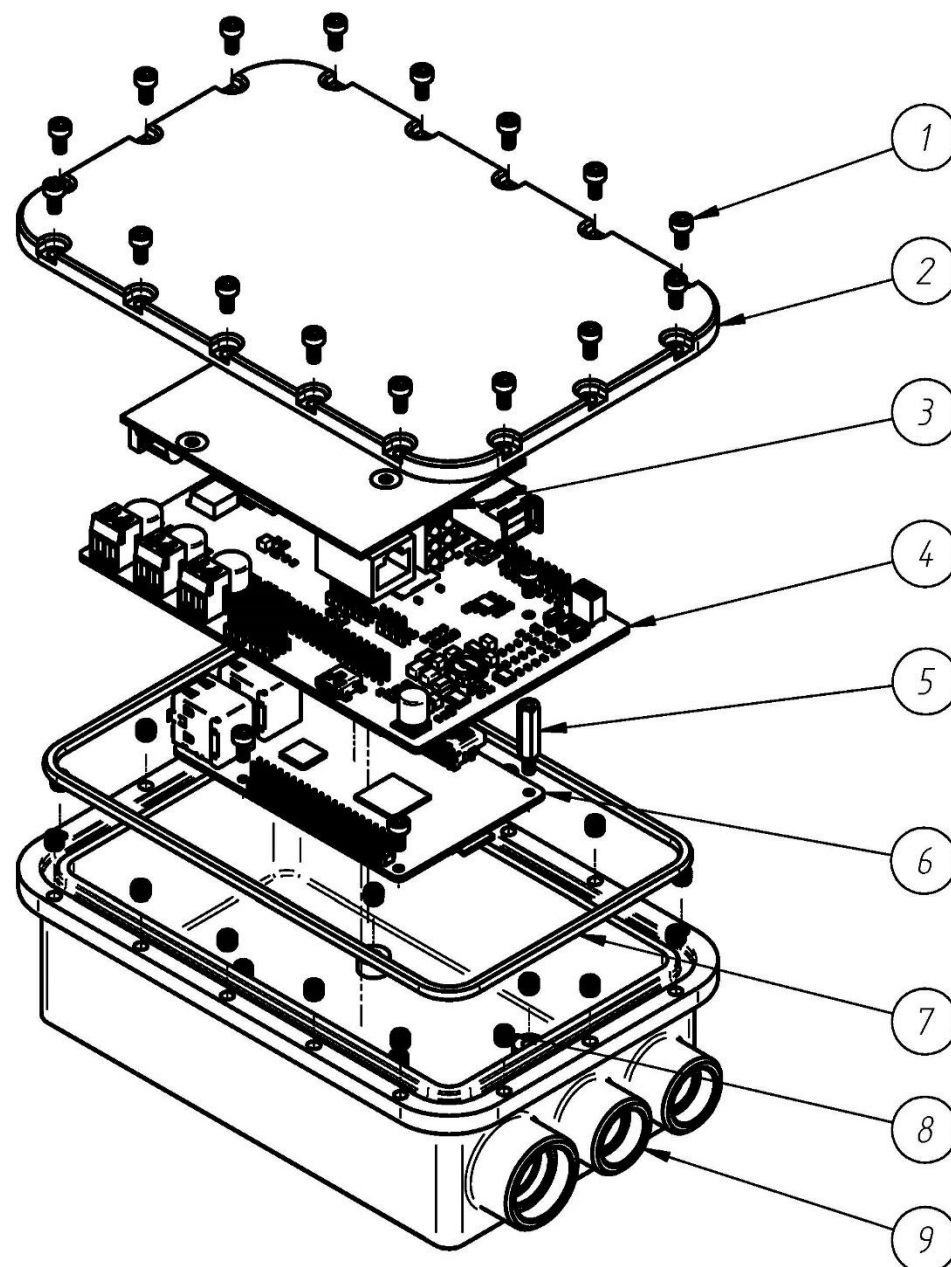
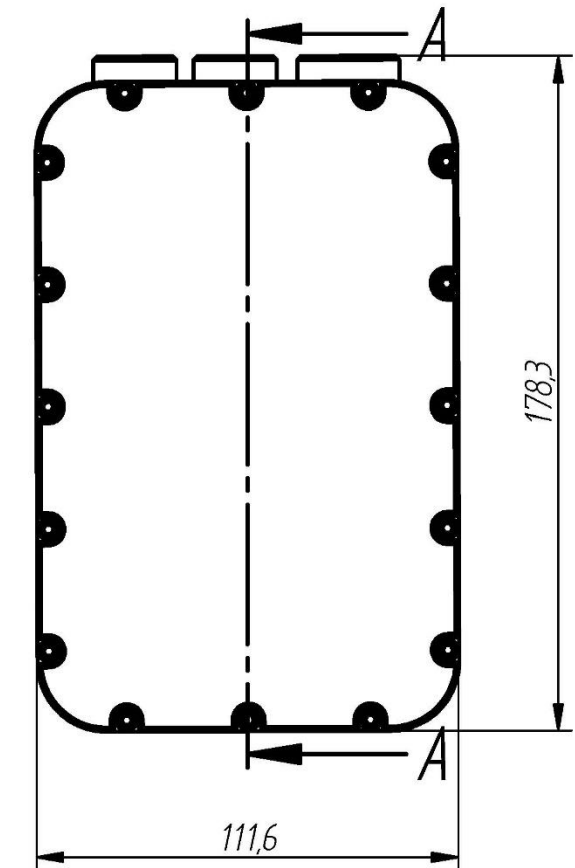
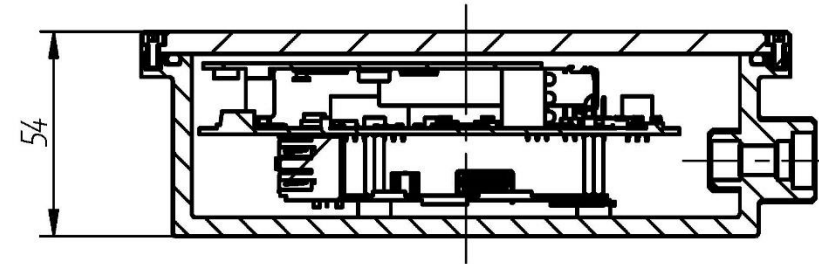
1. Märkimata raadiused 0,25 mm

	Materjal: <i>Formfutura EasyFil HIPS</i>	Näitamata piirhálbed: <i>ISO 2768</i>	Mass: <i>0.019</i>	Mõõt: <i>1:1</i>
Teostas	üliõp. Margus Nurme	Akupaki korpuse kaas		
Kontrollis	dotsent Veli Palge			
Kinnitas				
<b>Eesti Maaülikool</b> Estonian University of Life Sciences Tartu Tehnikakolledž Tartu Technology College		Leht: <i>1</i>	Tahis: <i>TN 18/120461 A 05 02 D</i>	




	<b>Materjal:</b> <i>Formfutura EasyFil HIPS</i>	<b>Näitamata piirhälbed:</b> <i>ISO 2768</i>	<b>Mass:</b> <i>0.026</i>	<b>Mööd:</b> <i>1:1</i>
<b>Teostas</b>	<i>üliõp. Margus Nurme</i>	<b>Nimetus:</b> <i>Akude pesa</i>		
<b>Kontrollis</b>	<i>dotsent Veli Palge</i>			
<b>Kinnitas</b>				
		<b>Leht:</b> <i>1</i>	<b>Tähis:</b> <i>TN 18/120461 A 05 03 D</i>	

VAADE A-A




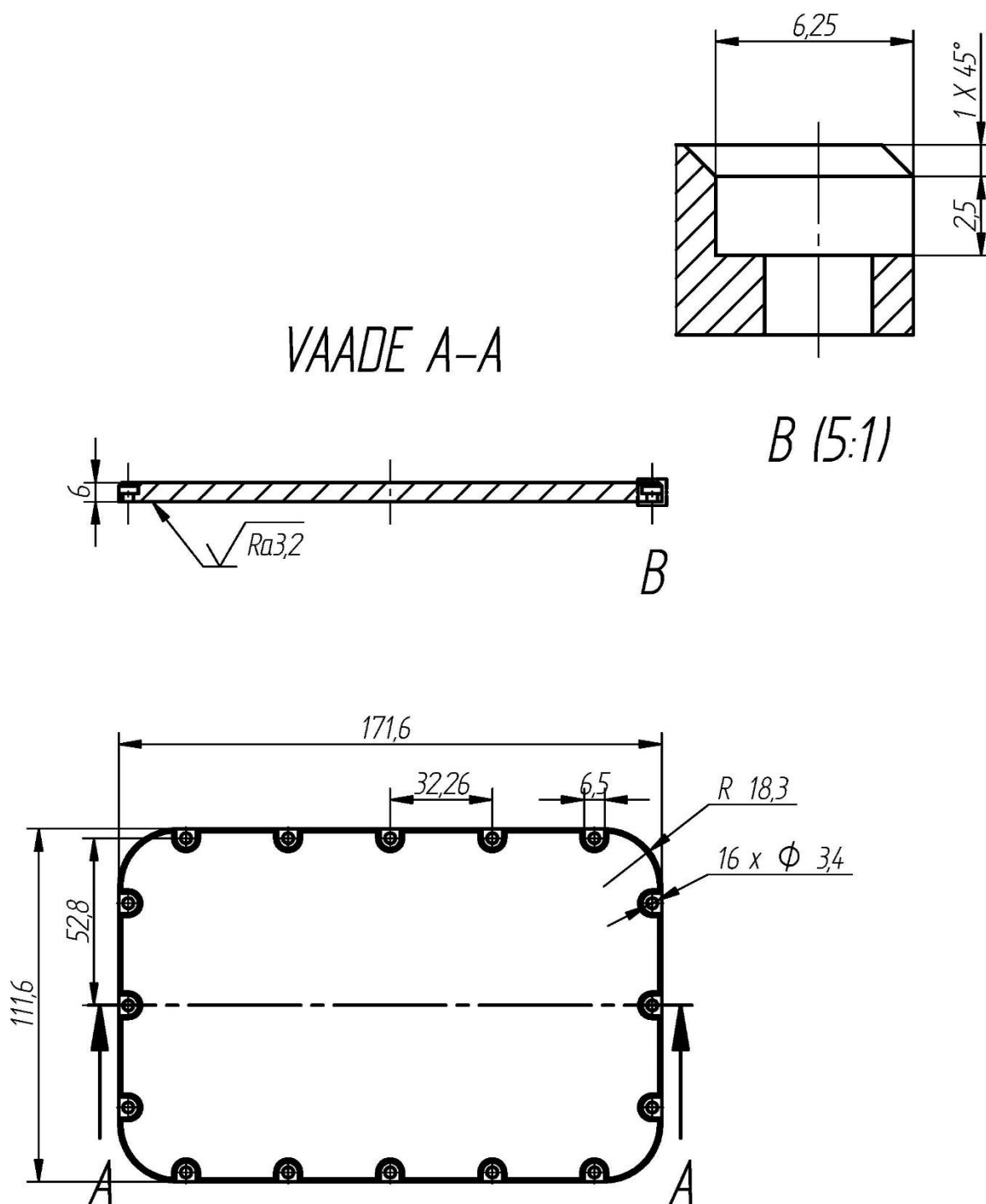
1. Pärast kaablite läbiviimist läbiviikude sisse valatud epoksiidliim Loctite EA 3430

1		Sisekuuskantpolt M3x 0.5-6, roostevaba teras 316		20	
2		Elektroonikakambri korpuse kaas, HIPS	TN 18/120461 A 06 02 D	1	
3		Ethernetist kiudoptiliseks meediakonverter		1	
4		Kontroller		1	
5		16mm distantspuks M3x 6 mm, roostevaba teras 316		2	
6		Raspberry Pi 3 mudel B		1	
7		O-rõngas nöö 3.53 mm 70A, nitriilkumm		1	
8		Tõmbemutter M3x 0.5-4, messing		20	Kuumpressitav
9		Elektroonikakambri korpus, HIPS	TN 18/120461 A 06 01 D	1	
Osa	Väl	Nimetus, materjal	Tähis	Hulk	Märkus
		Materjal:	Näitamata piirhálbed: ISO 2768	Mass: 0.553	Mõõt: 1:2
Teostas	üliõp. Margus Nurme		Elektroonikakamber		
Kontrollis	dotsent Veli Palge				
Kinnitas					
 <b>Eesti Maaülikool</b> Estonian University of Life Sciences Tartu Tehnikakolledž Tartu Technology College			Leht: 1	Tähis: TN 18/120461 A 06 00 K	

SOLID EDGE ACADEMIC COPY



	Materjal: <i>Formfutura EasyFil HIPS</i>	Näitamata piirhälbed: <i>ISO 2768</i>	Mass: <i>0.217</i>	Mõõt: <i>1:2</i>
Teostas	Üliõp. Margus Nurme	Nimetus:  <i>Elektroonikakambri korpus</i>		
Kontrollis	dotsent Veli Palge			
Kinnitas				
 <b>Eesti Maaülikool</b> Estonian University of Life Sciences Tartu Tehnikakolledž Tartu Technology College		Leht: <i>1</i>	Tähis: <i>TN 18/120461 A 06 01 D</i>	



	Materjal: Formfutura EasyFil HIPS	Näitamata piirhálbed: ISO 2768	Mass: 0.115	Mööõt: 1:2
Teostas	üliõp. Margus Nurme	Elektroonikakambri korpuse kaas		
Kontrollis	dotsent Veli Palge			
Kinnitas				
	Eesti Maaülikool Estonian University of Life Sciences Tartu Tehnikakolledž Tartu Technology College	Leht: 1	Tahis: TN 18/120461 A 06 02 D	

**LISA B – Arvutused**

## 1. Rõhu erinevus 50m sügavusel

### 1.1. Rõhu erinevus veepinnal ja 50 meetri sügavusel

$$\rho_{\text{vesi}} := 1000 \frac{\text{kg}}{\text{m}^3} \quad g = 9.807 \cdot \frac{\text{m}}{\text{s}^2} \quad h := 50\text{m}$$

$$P_{\text{H}_2\text{O}50\text{m}} := \rho_{\text{vesi}} \cdot g \cdot h$$

kus,  $\rho_{\text{vesi}}$  - vee tihedus,  $g$  - gravitatsioonikonstant ja  $h$  - vedelikusamba kõrgus.

$$P_{\text{H}_2\text{O}50\text{m}} = 490.332 \cdot \text{kPa}$$

Kuna ka õhukindlate korpuste sees on atmosfääri rõhk, siis 490 kPa on rõhu erinevus välispindade ja korpuste sisu vahel.

## 2. Allveesõiduki keskmine tihedus

### 2.1. Esialgne allveesõiduki keskmine tihedus

Allveesõiduki keskmise tiheduse leiame valemiga:

$$\rho_{\text{ROV}} = \frac{m_{\text{ROV}}}{V_{\text{ROV}}}$$

kus  $\rho_{\text{ROV}}$  - allveesõiduki keskmine tihedus,  $m_{\text{ROV}}$  - allveesõiduki mass,  $V_{\text{ROV}}$  - allveesõiduki ruumala.

$$V_{\text{ROV}} := 2754335.898 \text{mm}^3 = 2.754 \times 10^{-3} \cdot \text{m}^3 \quad m_{\text{ROV}} := 3.206 \text{kg}$$

$$\rho_{\text{ROV}} := \frac{m_{\text{ROV}}}{V_{\text{ROV}}} = 1163.983 \cdot \frac{\text{kg}}{\text{m}^3}$$



Lisa Bjärg

## 2.2. Allveesõiduki keskmine tihedus pärast Divinycell vahtplokki ja trimmimis raskuste lisamist

$$V_{\text{ROV}} := 3494346.179 \text{ mm}^3 = 3.494 \times 10^{-3} \cdot \text{m}^3 \quad m_{\text{ROV}} := 3.436 \text{ kg}$$

$$\rho_{\text{ROV}} := \frac{m_{\text{ROV}}}{V_{\text{ROV}}} = 983.303 \cdot \frac{\text{kg}}{\text{m}^3}$$

## 3. Allveesõidukile vajaminev ujukmaterjali kogus

### 3.1. Esialgne ujukmaterjali kogus

$$\rho_{\text{Divinycell}} := 100 \frac{\text{kg}}{\text{m}^3}$$

$$\rho_{\text{vesi}} := 1000 \frac{\text{kg}}{\text{m}^3}$$

$$m_{\text{H}_2\text{O}} := 2.845 \text{ kg}$$

$$m_{\text{ROV}} := 3.206 \text{ kg}$$

$$F_{\text{üleslükke}} := m_{\text{H}_2\text{O}} \cdot g = 27.9 \text{ N}$$

$$F_{\text{raskus}} := m_{\text{ROV}} \cdot g = 31.44 \text{ N}$$

Arvutame puuduoleva üleslükkejõu koos 4% varuga

$$F_{\text{puudu}} := \frac{F_{\text{raskus}} \cdot 4}{100} + (F_{\text{raskus}} - F_{\text{üleslükke}}) = 4.798 \text{ N}$$

$$m_{\text{H}_2\text{O\_puudu}} := \frac{F_{\text{puudu}}}{g} = 0.489 \text{ kg}$$

Arvutame vajaminev Divinycell vahtplokki ruumala 4% üleslükkejõu varuga, arvestades vahtplokki enda massiga

Given

$$x = V_{\text{Divinycell}} \cdot \rho_{\text{vesi}}$$

$$y = V_{\text{Divinycell}} \cdot \rho_{\text{Divinycell}} \quad x - y = 0.489 \text{ kg}$$

Lisa Bjärg

$$V_{\text{Divinycell}} \cdot \rho_{\text{vesi}} - V_{\text{Divinycell}} \cdot \rho_{\text{Divinycell}} = 0.489 \text{ kg solve, } V_{\text{Divinycell}} \rightarrow 0.00054$$

$$V_{\text{Divinycell}} := 0.000543 \text{ m}^3 = 543 \cdot \text{cm}^3 \quad m_{\text{H}_2\text{O\_vaht}} := 0.543 \text{ kg}$$

kus

$V_{\text{Divinycell}}$  - vajamineva Divinycell vahtploki ruumala

$\rho_{\text{Divinycell}}$  - Divinycell vahtploki tihedus

$m_{\text{H}_2\text{O}}$  - allveesõiduki poolt veest kõrvale tõrjutud vee mass

$m_{\text{ROV}}$  - allveesõiduki mass

$F_{\text{üleslükke}}$  - allveesõidukile vees mõjuv üleslükkejõud

$F_{\text{raskus}}$  - allveesõidukile mõjuv raskusjõud

$F_{\text{puudu}}$  - puuduoleva üleslükkejõu suurus

$m_{\text{H}_2\text{O}}$  - puuduolev allveesõiduki poolt veest kõrvale tõrjutud vee mass

$m_{\text{H}_2\text{O\_vaht}}$  - Divinycell vahu lisamisest allveesõiduki poolt veest kõrvale tõrjutud massile juurde tulnud vee mass

$m_{\text{H}_2\text{O\_puudu}}$  - puuduolev allveesõiduki poolt eemale tõrjutud vee hulga mass allveesõiduki massi ületamiseks

Antud vahu lisamisel ületab allvee sõiduki poolt välja tõrjutud vee massi pärast raskusjõu ületamist 182 g ja pärast 165 g trimmimis raskuste lisamist 17 g võrra

$$m_{\text{trimmimis\_raskus}} := 0.165 \text{ kg}$$

Leiame kui palju veest kõrvale tõrjutud vee mass ületab allveesõiduki massi pärast trimmimis raskuste paigaldamist:

$$m_{\text{H}_2\text{O\_jääk}} := m_{\text{H}_2\text{O}} + m_{\text{H}_2\text{O\_vaht}} - m_{\text{ROV}} - m_{\text{trimmimis\_raskus}} = 0.017 \text{ kg}$$

### 3.2. Täiendav ujukmaterjali kogus

Täiendava üleslükkejõu saamiseks lisati veel 107 cm<sup>3</sup> Divinycell H vahtu, pärast mida ületab vee poolt tekitatav üleslükkejõud allveesõidukile mõjuva raskusjõu ligikaudu 1,7 % võrra.

Lisa Bjärg

Täiendavalt lisatud Divinycell H vahu ruumala:

$$V_{\text{Divinycell\_lisa}} := 107 \text{ cm}^3$$

Täiendavalt lisatud Divinycell H mass:

$$m_{\text{H}_2\text{O\_vaht\_lisa}} := V_{\text{Divinycell\_lisa}} \cdot \rho_{\text{Divinycell}} = 0.0107 \text{ kg}$$

Lisatud Divinycell vahu ruumala kokku:

$$V_{\text{Divinycell\_vaht\_kokku}} := V_{\text{Divinycell}} + V_{\text{Divinycell\_lisa}} = 650 \cdot \text{cm}^3$$

Lisatud Divinycell vahu mass kokku:

$$m_{\text{Divinycell\_vaht\_kokku}} := V_{\text{Divinycell\_vaht\_kokku}} \cdot \rho_{\text{Divinycell}} = 0.065 \text{ kg}$$

Allveesõidkui poolt kõrvale tõrjutud vee mass kokku pärast Divinycell H vahu lisamist:

$$m_{\text{H}_2\text{O\_vahuga\_kokku}} := m_{\text{H}_2\text{O}} + m_{\text{H}_2\text{O\_vaht}} + m_{\text{H}_2\text{O\_vaht\_lisa}} = 3.399 \text{ kg}$$

Allveesõiduki mass pärast Divinycell H vahu ja trimmimis raskuste lisamist:

$$m_{\text{ROV\_kokku}} := m_{\text{ROV}} + m_{\text{trimmimis\_raskus}} + m_{\text{Divinycell\_vaht\_kokku}} = 3.436 \text{ kg}$$

Massidele vastavad jõud:

$$F_{\text{üleslükke\_kokku}} := m_{\text{H}_2\text{O\_vahuga\_kokku}} \cdot g = 33.33 \text{ N}$$

$$F_{\text{raskusjõud\_kokku}} := m_{\text{ROV\_kokku}} \cdot g = 33.696 \text{ N}$$

Protsentuaalselt kui palju allveesõidukile mõjuv ülelülkejõud ületab talle mõjuva raskusjõu:

$$F_{\text{ülejääk}} := \frac{34.274 \cdot 100}{33.696} - 100 = 1.715\%$$

## 4. Allveesõiduki kaabli keskmine tihedus

### 4.1. Esialgne valguskaabli keskmine tihedus

$$l_{\text{kaabel}} := 100\text{m} \quad d_{\text{kaabel}} := 1.8\text{mm} = 1.8 \times 10^{-3}\text{m} \quad m_{\text{kaabel}} := 0.348\text{kg}$$

Valguskaabli keskmise tiheduse leiame valemiga:

$$\rho_{\text{kaabel}} = \frac{m_{\text{kaabel}}}{V_{\text{kaabel}}}$$

kus  $\rho_{\text{kaabel}}$  - kaabli tihedus,  $m_{\text{kaabel}}$  - kaabli mass,  $V_{\text{kaabel}}$  - kaabli ruumala.

Valguskaabli ruumala leiame valemiga:

$$V_{\text{kaabel}} = S_{\text{kaabel}} \cdot l_{\text{kaabel}}$$

kus  $S_{\text{kaabel}}$  - kaabli ristlõike pindala ja  $l_{\text{kaabel}}$  - kaabli pikkus.

Valguskaabli ristlõike pindala leiame valemiga:

$$S_{\text{kaabel}} = \pi \cdot \left( \frac{d_{\text{kaabel}}}{2} \right)^2 \quad S_{\text{kaabel}} := \pi \cdot \left( \frac{d_{\text{kaabel}}}{2} \right)^2 = 2.545 \times 10^{-6} \text{m}^2$$

kus  $d_{\text{kaabel}}$  - kaabli ristlõike läbimõõt.

$$V_{\text{kaabel}} := S_{\text{kaabel}} \cdot l_{\text{kaabel}} = 2.545 \times 10^{-4} \cdot \text{m}^3$$

$$\rho_{\text{kaabel}} := \frac{m_{\text{kaabel}}}{V_{\text{kaabel}}} = 1367.554 \cdot \frac{\text{kg}}{\text{m}^3}$$

Lisa Bjärg

## 4.2. Valguskaabli keskmine tihedus pärast täiendamist

$$l_{\text{kaabel}} := 100\text{m} \quad d_{\text{kaabel}} := 2.95\text{mm} = 2.95 \times 10^{-3}\text{m} \quad m_{\text{kaabel}} := 0.664\text{kg}$$

$$S_{\text{kaabel}} = \pi \cdot \left( \frac{d_{\text{kaabel}}}{2} \right)^2 \quad S_{\text{kaabel}} := \pi \cdot \left( \frac{d_{\text{kaabel}}}{2} \right)^2 = 6.835 \times 10^{-6}\text{m}^2$$

$$V_{\text{kaabel}} := S_{\text{kaabel}} \cdot l_{\text{kaabel}} = 6.835 \times 10^{-4} \cdot \text{m}^3$$

$$\rho_{\text{kaabel}} := \frac{m_{\text{kaabel}}}{V_{\text{kaabel}}} = 971.481 \cdot \frac{\text{kg}}{\text{m}^3}$$

## 5. Allveesõidukile liikumisel mõjuv takistus vees

Vee all liikuvale allveesõidukile mõjuv takistus on leitav valemiga:

$$R = \frac{1}{2} \cdot \sigma \cdot A \cdot V^2 \cdot C_d$$

kus  $\sigma$  - vedeliku tihedus/gravitatsioonikonstant,  $A$  - allveesõiduki esiotsa pindala,  $V$  - liikumiskiirus ja  $C_d$  - mittedimensionaalne esiotsa takistuse koefitsient

### 5.1. Sõidukile kiirusel 0.5 m/s vastu mõjuv jõud

$$\sigma := \frac{1000 \frac{\text{kg}}{\text{m}^3}}{\text{g}} \quad A_s := 22729.51\text{mm}^2 = 0.023\text{m}^2$$

$$V_s := 0.5 \frac{\text{m}}{\text{s}} \quad C_{ds} := 0.8$$

$$R_s := \frac{1}{2} \cdot \sigma \cdot A_s \cdot V_s^2 \cdot C_{ds} = 0.232\text{kg} \quad F_s := R_s \cdot g = 2.273\text{N}$$

Lisa Bjärg

## 5.2. Sõidukile kiirusel 1 m/s vastu mõjuv jõud

$$V_s := 1 \frac{\text{m}}{\text{s}}$$

$$R_s := \frac{1}{2} \cdot \sigma \cdot A_s \cdot V_s^2 \cdot C_{ds} = 0.927 \text{ kg} \quad F_s := R_s \cdot g = 9.092 \text{ N}$$

## 6. Allveesõiduki liikumisel kaablile mõjuv takistus

### 6.1. Kaablile kiirusel 0.5 m/s vastu mõjuv jõud eeldusel, et 50 m kaablit on vees

Allveesõiduki kaablite puhul leitakse tegur A valemiga:

$$A_k = d \cdot l$$

kus d - valguskaabli läbimõõt ja l - kaabli pikkus

$$d := 2.95 \text{ mm} = 2.95 \times 10^{-3} \text{ m} \quad l := 50 \text{ m} \quad \sigma := \frac{1000 \frac{\text{kg}}{\text{m}^3}}{g}$$

$$A_k := d \cdot l = 0.148 \text{ m}^2 \quad C_{dk} := 1.2 \quad V_k := 0.5 \frac{\text{m}}{\text{s}}$$

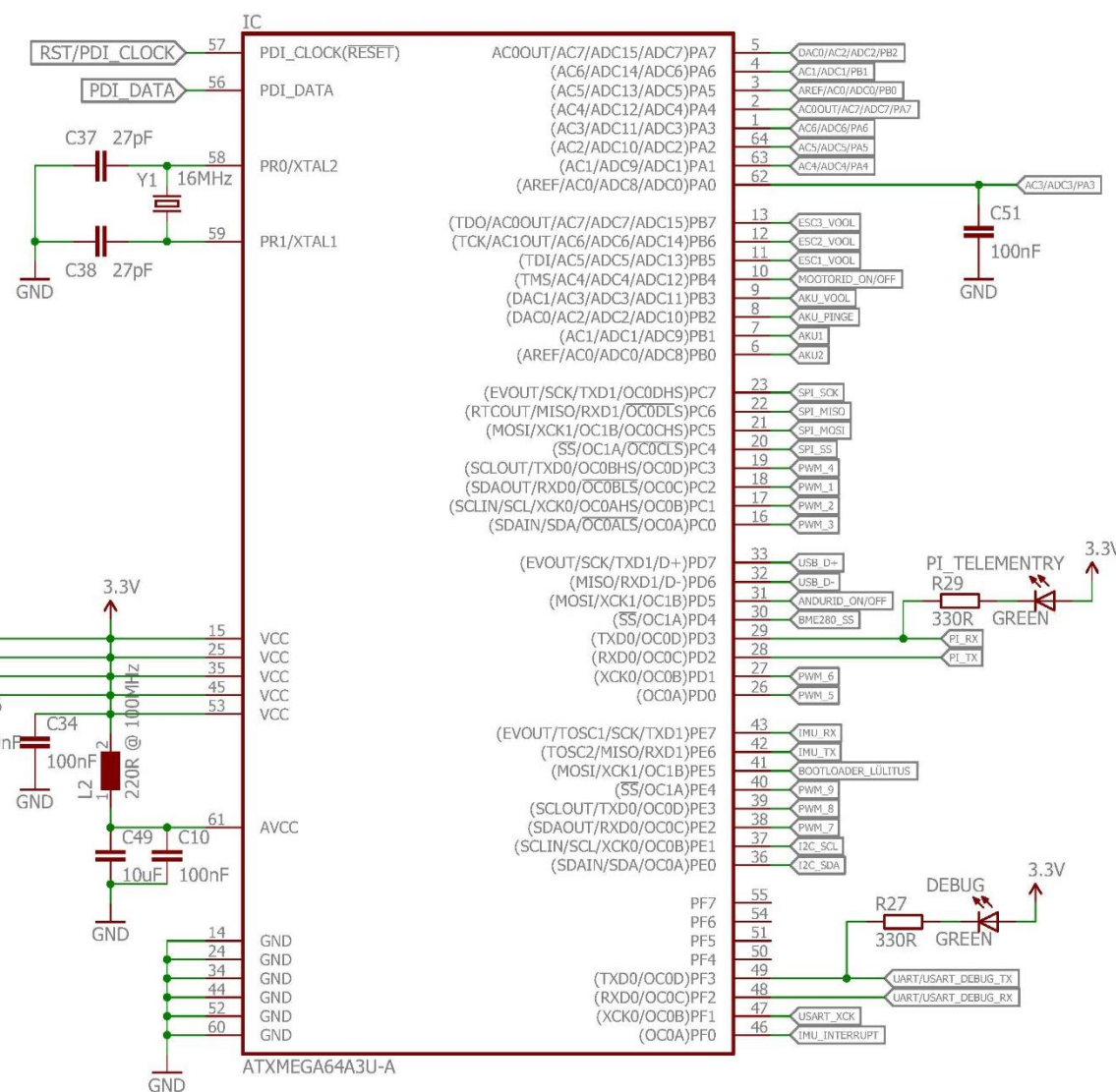
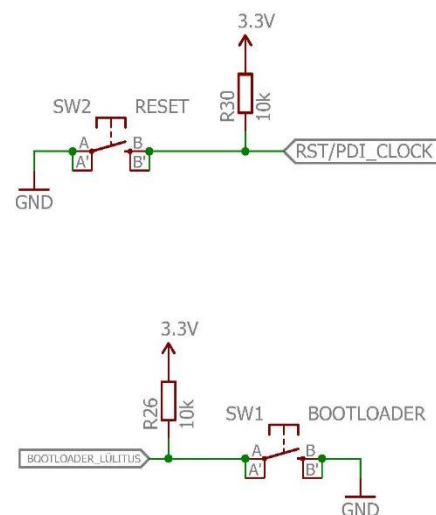
$$R_k := \frac{1}{2} \cdot \sigma \cdot A_k \cdot V_k^2 \cdot C_{dk} = 2.256 \text{ kg} \quad F_s := R_k \cdot g = 22.125 \text{ N}$$

### 6.2. Kaablile kiirusel 1 m/s vastu mõjuv jõud

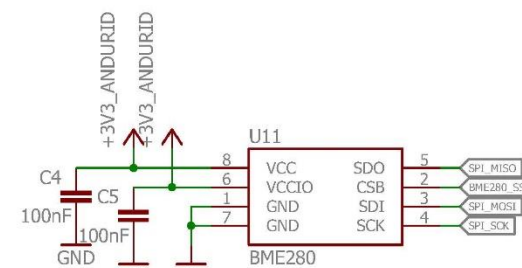
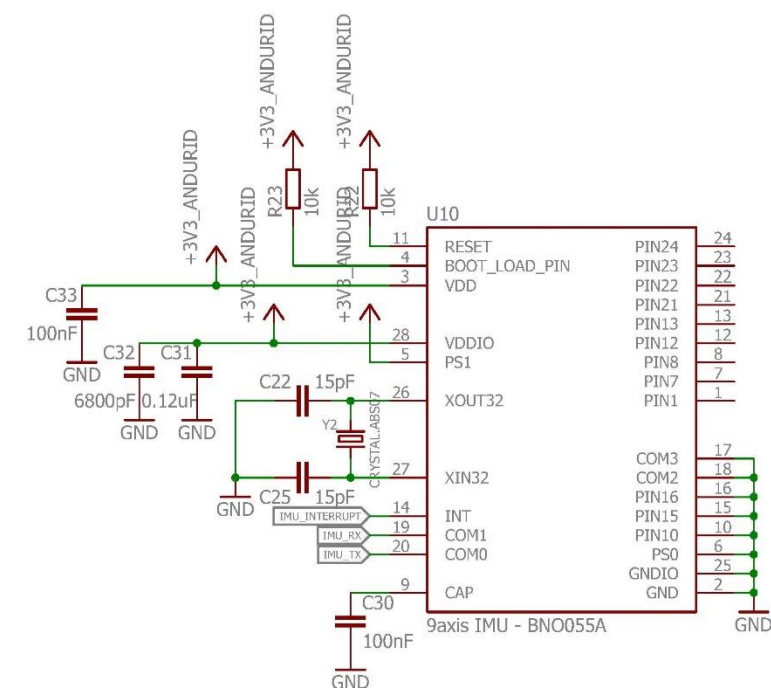
$$V_k := 1 \frac{\text{m}}{\text{s}}$$

$$R_k := \frac{1}{2} \cdot \sigma \cdot A_k \cdot V_k^2 \cdot C_{dk} = 9.024 \text{ kg} \quad F_s := R_k \cdot g = 88.5 \text{ N}$$

## **LISA C – Kontrollerplaadi elektroonikaskeem**

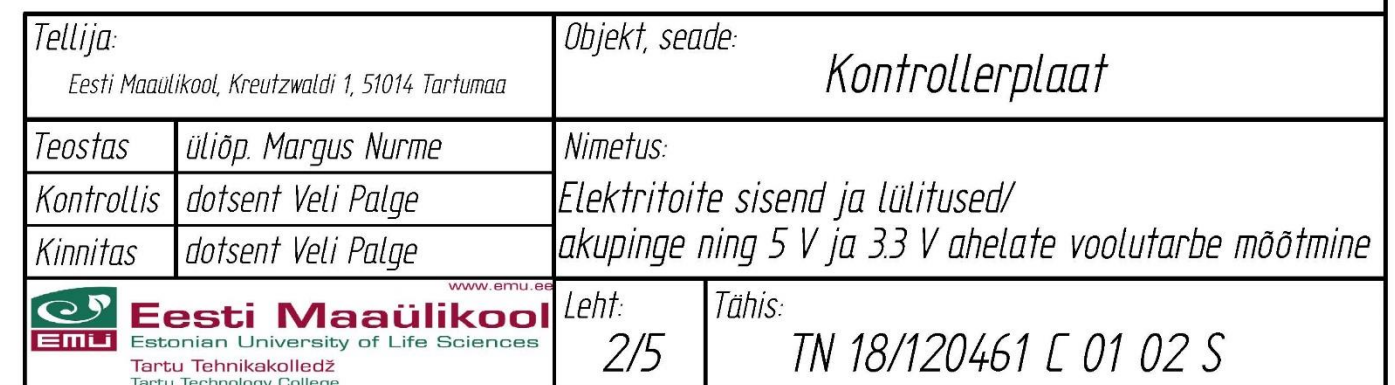


RPI kinnitus

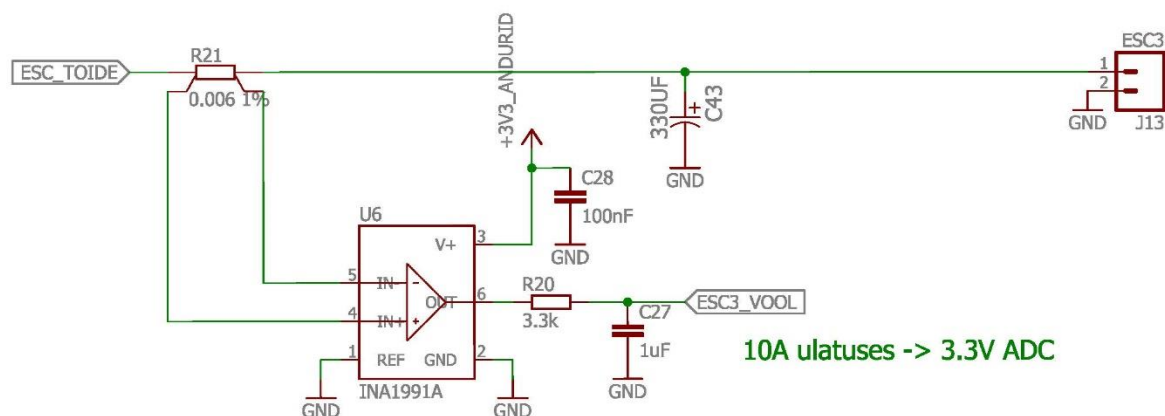


Tellija:		Objekt, seade:	
Eesti Maaülikool, Kreutzwaldi 1, 51014 Tartumaa		Kontrollerplaat	
Teostas	üliõp. Margus Nurme	Nimetus:	
Kontrollis	dotsent Veli Palge		
Kinnitas	dotsent Veli Palge		
Leht:		Tähis:	
1/5		TN 18/120461 C 01 01 S	

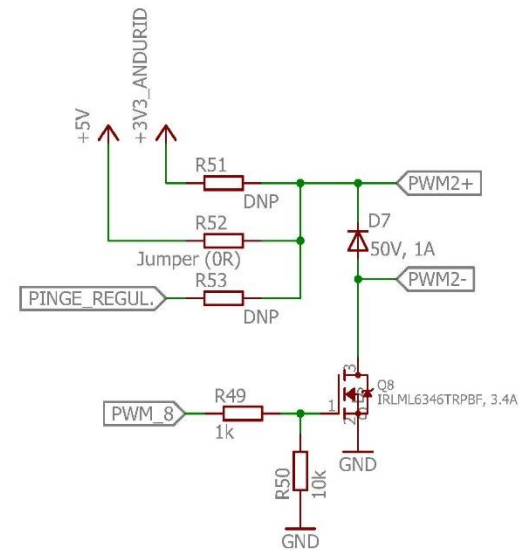
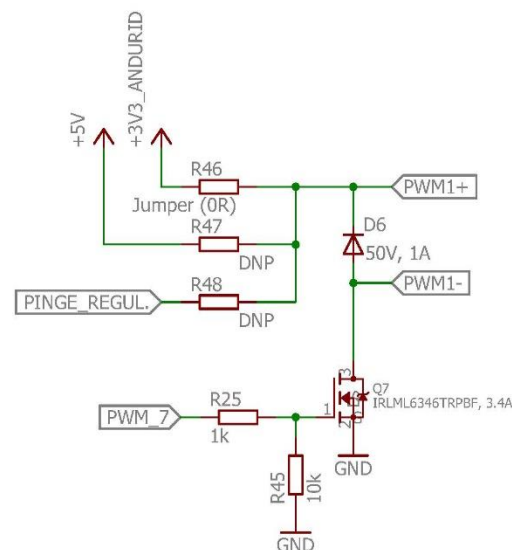
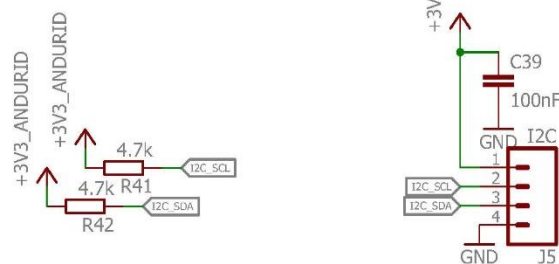
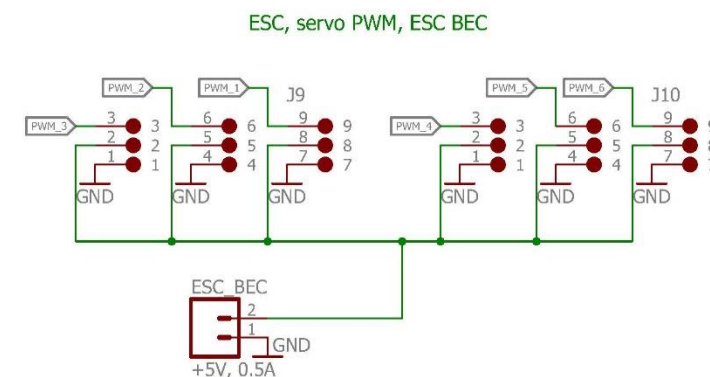
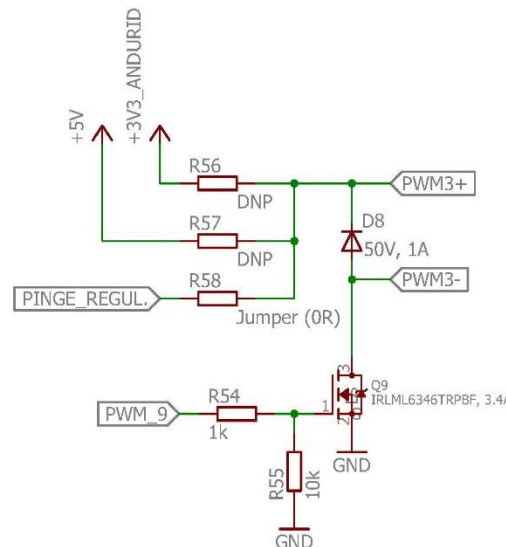
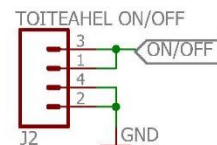
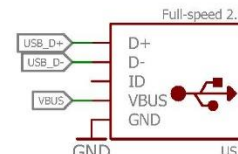
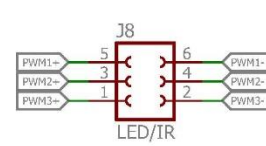
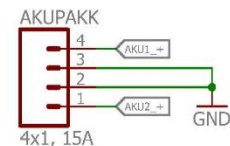
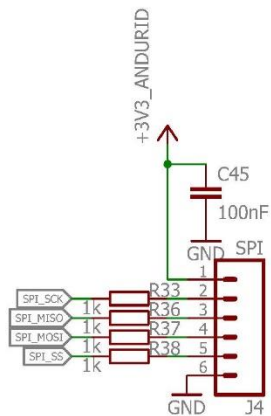
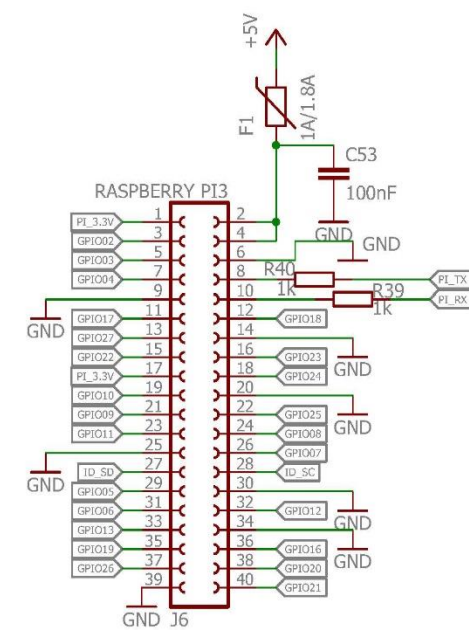
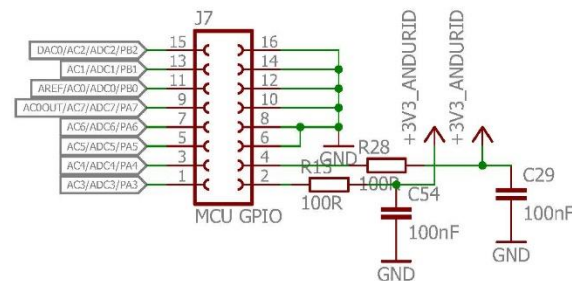
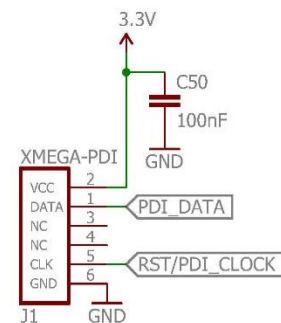
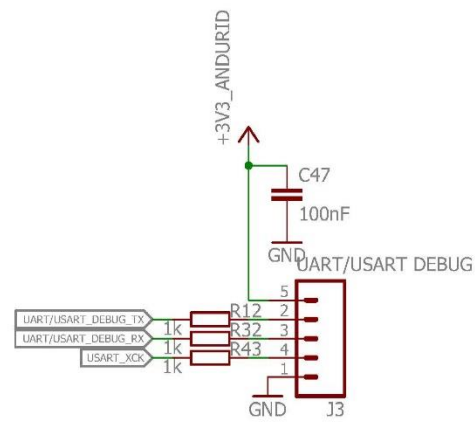








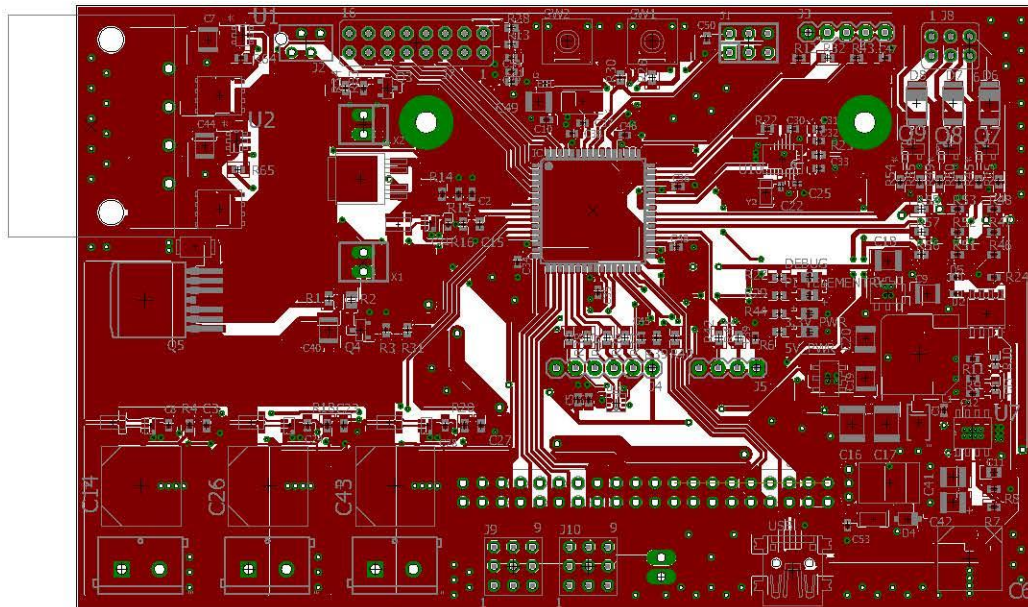
TN 18/120461 [ 01 04 S



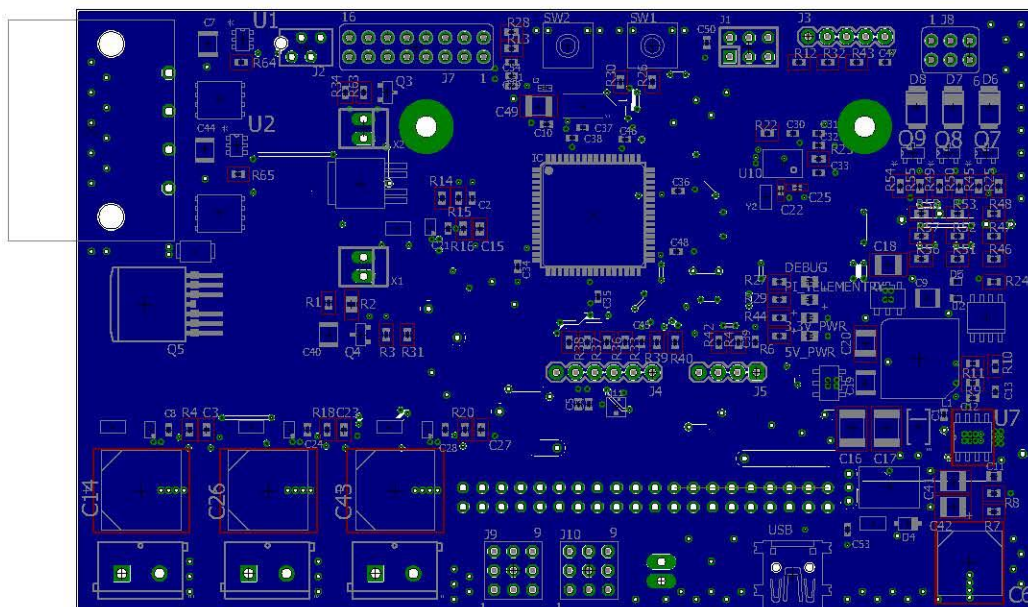
Tellija: Eesti Maaülikool, Kreutzwaldi 1, 51014 Tartumaa		Objekt, seade: Kontrollerplaat	
Teostas	üliõp. Margus Nurme	Sisendid ja väljundid	
Kontrollis	dotsent Veli Palge		
Kinnitas	dotsent Veli Palge		
Leht: 5/5		Tähis: TN 18/120461 C 01 05 S	



## Ülemine pool



## Alumine pool



**LISA D – Mikrokontrolleri püsivara**

```

#define F_CPU 32000000UL

#include <stdio.h>

#include <avr/io.h>

#include <string.h>

#include <inttypes.h>

#include <util/delay.h>

#include <avr/interrupt.h>

#include <avr/ring_buffer.h>

//Algaväärtustame ja arvutame vastavalt mootoritele seatud v6imsuse astmele nende kiirused
#define MOTOR_NEUTRAL 1500

//"boolean" andmetüübid

#define True 1

#define False 0

typedef unsigned char bool;

bool a = True;

bool b = False;

static bool fillrxBuffer = False;

static volatile bool storeString = False; //terve käsklus on üle UART sisse loetud

static bool commandReady = False; //käskuls rx puhvril edukalt käsu puhvrilisse loetud

const char cmdStartChar = '<'; //käsu alguse tähis

const char endChar = ';'; //käsu lõpu tähis

volatile uint16_t period = 0;

Lisa C järg

```

Lisa C järg

```
static volatile buffer_t rxBuffer; //puhver saabuvate käskude sisse lugemiseks
```

```
char commandBuffer[BUFFER_SIZE + 1]; //puhver käskude töötlemiseks
```

```
int BufferIsEmpty(buffer_t *p)
```

```
{
```

```
    //Tagasta TRUE kui etteantud puhver on tühi
```

```
    return (p->rear + 1) % (BUFFER_SIZE + 1) == p->front;
```

```
}
```

```
int BufferIsFull(buffer_t *p)
```

```
{
```

```
    //Tagasta TRUE kui etteantud puhver on täis
```

```
    return (p->rear + 2) % (BUFFER_SIZE + 1) == p->front;
```

```
}
```

```
//Puhvri initsialiseerimine
```

```
void BufferInit(buffer_t *p)
```

```
{
```

```
    p->front = 1;
```

```
    p->rear = 0; //puhvri koht, kuhu lisada
```

```
}
```

```
//char sisestamine määratud puhvri sabasse
```

```
int BufferInsert(buffer_t *p, uint8_t byte)
```

```
{
```

```
    if ((p->rear + 2) % (BUFFER_SIZE + 1) == p->front)
```

```
        return FALSE;
```

```
    p->rear = (p->rear + 1) % (BUFFER_SIZE + 1);
```



Lisa C järg

```
p->buffer[p->rear] = byte;

    return TRUE;
}

//Eemalda ja tagasta char määratud puhvri algusest
uint8_t BufferRemove(buffer_t *p)
{
    uint8_t byte;

    byte = p->buffer[p->front];

    p->front = (p->front + 1) % (BUFFER_SIZE + 1);

    return byte;
}

//Tagastada char määratud puhvri algusest
uint8_t BufferFront(buffer_t *p)
{
    return p->buffer[p->front];
}

//Kustuta char määratud puhvri algusest
int BufferDelete(buffer_t *p)
{
    if ((p->rear + 1) % (BUFFER_SIZE + 1) == p->front)

        return FALSE;

    p->front = (p->front + 1) % (BUFFER_SIZE + 1);

    return TRUE;
}
```

Lisa C järg

```
void ESC_PWM_init(uint16_t PWM_sagedus, uint32_t cpu_MHz) //PWM_sagedus -
soovitatav PWM signaali sagedus hertsides; cpu_MHz - kasutatav cpu sagedus megahertsides
{
    uint32_t cpu_sagedus = cpu_MHz * 1000000; //Teisendame MHz->Hz

    uint16_t prescaler, v2ikseim_jagatav;

    PORTC.DIR |= PIN0_bm | PIN1_bm | PIN2_bm; //Port C pin 0 v2ljundiks. m22rab
    TTC0.taimer (kalibreerimis väljund)

    TCC0.CTRLD &= ~PIN0_bm; //Keelata (nulli) event action PORT C 0 taimeritel

    TCC0.CTRLFCLR &= ~PIN0_bm; //loendur yles lugemal

    //Arvutada soovitud PWM sagedusele CPU sageduest s6ltuvalt v2iseim jagaja
    v2ikseim_jagatav = cpu_sagedus / PWM_sagedus / 0xFFFF; // 0xFFFF = 65535

    TCC0.CTRLA = 0x00; //Seame k6ik bitid maha (kui varasemalt on sagedusjagur juba
seatud)

    TCC0.PER = 0x00; //Seame perioodi 0, kui varasemalt on periood seatud

    if (v2ikseim_jagatav < 1) {
        prescaler = 1;

        TCC0.CTRLA |= PIN0_bm; //Seame kellaks cpu_sagedus/sagedusjagur 1 jne
    } else if (v2ikseim_jagatav < 2) {
        TCC0.CTRLA |= PIN1_bm;

        prescaler = 2;
    } else if (v2ikseim_jagatav < 4) {
        TCC0.CTRLA |= PIN0_bm | PIN0_bm;

        prescaler = 4;
    } else if (v2ikseim_jagatav < 8) {
```

Lisa C järg

```
TCC0.CTRLA |= PIN2_bm;

prescaler = 8;

} else if (v2ikseim_jagatav < 64) {

TCC0.CTRLA |= PIN0_bm | PIN2_bm;

prescaler = 64;

} else if (v2ikseim_jagatav < 256) {

TCC0.CTRLA |= PIN1_bm | PIN2_bm;

prescaler = 256;

} else {

TCC0.CTRLA = PIN0_bm | PIN1_bm | PIN2_bm;

prescaler = 1024;

}

//Arvuta periood vastavalt leitud sagedusjagurile
period = cpu_sagedus / (PWM_sagedus*prescaler)-1; //ca 50Hz
TCC0.PER = period-69; //Sea periood. 69 korrigeerimiseks

//Veenduda, et perioodi pikkus on 100 takti
if (period < 100) {

//Liiga lyhikese perioodi korral

TCC0.CTRLA |= 0x00; //Timer/counter OFF

TCC0.PER = 0x00; //Periood 0 - signaali ei tekitata

}

TCC0.CTRLB |= PIN0_bm | PIN1_bm | PIN4_bm | PIN5_bm | PIN6_bm; //Single-
slope PWM režiim; lubada Compare kanalil A, B ja C

}
```

Lisa C järg

//Kanal CCA - vertikaalne mootor; kanal CCB - tüürpoordi mootor; kanal CCC - pakpoordi mootor

//Vertikaalse mootori täitetegur; microseconds - seatav täitetegur mikrosekundites

void Vertical\_thruster(int microseconds)

```
{  
    uint16_t CCA_duty_cycle;  
    CCA_duty_cycle = microseconds/2;  
    TCC0.CCA = CCA_duty_cycle;  
}
```

//Tüürpoord

void Starboard\_thruster(int microseconds)

```
{  
    uint16_t CCB_duty_cycle;  
    CCB_duty_cycle = microseconds/2;  
    TCC0.CCB = CCB_duty_cycle;  
}
```

//Pakpoord

void Port\_thruster(int microseconds)

```
{  
    uint16_t CCC_duty_cycle;  
    CCC_duty_cycle = microseconds/2;  
    TCC0.CCC = CCC_duty_cycle;  
}
```

//Mootorid töövalmis (1500 täitetegur)

Lisa C järg

```
void Motors_standby()
{
    Vertical_thruster(MOTOR_NEUTRAL);
    Starboard_thruster(MOTOR_NEUTRAL);
    Port_thruster(MOTOR_NEUTRAL);
}

void clk_setup()
{
    //Konfi kell 32MHz peale

    OSC.CTRL |= OSC_RC32MEN_bm | OSC_RC32KEN_bm; //32MHz ja 32KHz
    ostsillaatorid sisse

    while(!(OSC.STATUS & OSC_RC32KRDY_bm)); //Oota kuni 32KHz
    ostsillaator stabiliseerub

    while(!(OSC.STATUS & OSC_RC32MRDY_bm)); //Oota kuni 32Mhz
    ostsillaator stabiliseerub

    DFLLRC32M.CTRL = DFLL_ENABLE_bm ; //DFLL sisse (kalibreerida
    32Mhz ostsillaator 32KHz ostsillaatori järgi)

    CCP = CCP_IOREG_gc; //Turvalisus kella uuendamiseks v2lja

    CLK.CTRL = CLK_SCLKSEL_RC32M_gc; //Lülituda 32MHz kella peale

    OSC.CTRL &= ~OSC_RC2MEN_bm; //Lülita 2Mhz ostsillaator v2lja
}

void uart_setup()
{
    PORTD.OUT |= PIN3_bm;

    PORTD.DIR |= PIN3_bm;

    PORTD.DIR &= ~PIN2_bm;
```

Lisa C järg

```
//Baud rate 115200, Asynchronous, normal speed mode

USARTD0.BAUDCTRLB = (-3<<USART_BSCALE_gp); //BSCALE: -3

USARTD0_BAUDCTRLA = 131; //BSEL: 131

USARTD0_CTRLA |= USART_RXCINTLVL_HI_gc; //Lubada USART Receive
Complete katkestus

USARTD0.CTRLA |=
USART_CMODE_ASYNCHRONOUS_gc|USART_PMODE_DISABLED_gc|USART_CH
SIZE_8BIT_gc; //Asünkroone UART, 8 andme bitti, 1 stop bitt, ilma paarsuse kontrollita

USARTD0.CTRLB |= USART_TXEN_bm|USART_RXEN_bm; //Lubada andmete
vastuvõtmine ja saatmine
}

void uart_transmit()
{

    USARTD0_CTRLA |= USART_DREINTLVL_HI_gc; //Lubada USART Buffer
    Empty (Tx) katkestus
}

int main(void)
{

    cli(); //Keelata katkestused

    clk_setup(); //Taktsagedus 32MHz peale

    uart_setup(); //USART sisse

    ESC_PWM_init(50,32);

    PORTB.DIR |= PIN4_bm; //port B pin 4 v2ljundiks(mootorite toide)

    PORTB.OUT |= PIN4_bm; //Pingestame mootorid

    Motors_standby(); //Mootorid valmis seada

    BufferInit(&rxBuffer); //Initsialiseerime puhvri
```

Lisa C järg

```
PMIC.CTRL |= PMIC_LOLVLEN_bm | PMIC_MEDLVLEN_bm |
PMIC_HILVLEN_bm; //Lubada katkestused k6igil tasemetel

sei(); //Lubada katkestused

while (1)
{
    if (storeString == True) //Oota , et terve käsk oleks rx puhvris
    {
        uint8_t commandSize = 0;

        while (!BufferIsEmpty(&rxBuffer)) { //Kuni rx puhver pole tühi

            commandBuffer[commandSize++] =
BufferRemove(&rxBuffer); //Võta rxBuffer`ist element ja pane commandBuffrisse; seejärel
kustuta element rxBuffer`ist

            if (commandSize == commandBuffer) //Kui käsu suurus on
võrdne käsupuhvri suurusega, siis on käsk puhvri jaoks liiga suur (lõpus NULL
TERMINATOR)

                {

                    commandBuffer[sizeof(commandBuffer) - 1] = '\0';
//Hävita kogu käsk

                }

        }

        storeString = False; //rxBuffer on tühi

        commandReady = True; //andmed rx puhvrist käsupuhvrissse viidud

        if (commandReady == True)
        {

            //Käsupuhvris oleva käsu käsitlemine

            if (commandBuffer[0] == '<' && commandBuffer[1] == 'M')
            {
```

## Lisa C järg

```
uint16_t cmd_port;

uint16_t cmd_starboard;

uint16_t cmd_vertical;

uint16_t calculated_checksum;

uint16_t checksum;


cmd_port = commandBuffer[2] | commandBuffer[3] <<
8; //2 ja 3 baidi kokku täisarvuks - pakpoordi mootori käsk

cmd_starboard = commandBuffer[4] |
commandBuffer[5] << 8; //4 ja 5 baidi kokku täisarvuks - tüürpoordi mootori käsk

cmd_vertical = commandBuffer[6] | commandBuffer[7]
<< 8; //6 ja 7 baidi kokku täisarvuks - vertikaalse mootori käsk


calculated_checksum = cmd_port + cmd_starboard +
cmd_vertical;

checksum = commandBuffer[8] | commandBuffer[9] <<
8; //8 ja 9 bait kokku täisarvuks - Pi poolt arvutatud checksum

if (calculated_checksum == checksum)
{
    //Kui summad OK, saada käsud laiali
    Port_thruster(cmd_port);
    Starboard_thruster(cmd_starboard);
    Vertical_thruster(cmd_vertical);
}
else if (calculated_checksum != checksum)
{
    //Summad ei klappinud, eira käsku
```



Lisa C järg

```
    }  
    }  
    }  
    }  
    }  
}  
  
//USART receive katkestus  
ISR(USARTD0_RXC_vect)  
{  
    unsigned char Byte;  
    Byte = USARTD0.DATA;  
    if (Byte == cmdStartChar) //Kas saabus char '<' (0x3c)?  
    {  
        fillrxBuffer = True; //Lubada andmete paigutamine rxBuffrisse  
    }  
    if (fillrxBuffer == True)  
    {  
        if (Byte == endChar) //Kas saabus ';' (0x3b)?  
        {  
            fillrxBuffer = False; //Sulge rxBuffer kuni tuleb uuesti '<'  
            storeString = True; //käsk on koos ja töötlemiseks valmis  
            Byte = '\0'; //Lisame puhvrissse käsu lõppu NULL-TERMINATOR  
        }  
        BufferInsert(&rxBuffer, Byte); //Andmed puhvrissse
```

Lisa C järg

```
    }  
    if (BufferIsFull(&rxBuffer)) //Kui puhver on täis  
    {  
        memset(&rxBuffer, 0, BUFFER_SIZE + 1); //Pühi andmed rxBufferis.  
        fillrxBuffer = False; //Sulge rxBuffer kuni tuleb uuesti '<'  
        storeString = False;  
    }  
}
```

## **LISA E - pardaarvuti tarkvara**

Node server

//Laeme moodulid

```
var express = require('express'),  
    app = express(),  
    server = require('http').createServer(app),  
    io = require('socket.io').listen(server),  
    serialport = require('serialport');  
SerialPort = serialport.SerialPort;  
fs = require('fs'), //failide lugemiseks  
path = require('path'), //failide asukohtadega tegelemise moodul  
favicon = require('serve-favicon'), //moodul ikooni kuvamiseks  
logger = require('morgan'), //logimise moodul  
errorhandler = require('errorhandler'); //vigadega tegelemiseks  
vapix = require('vapix');
```

//Renderdusmootori seadestamine

```
app.set('views', path.join(__dirname + '/views'));  
app.set('view engine', 'ejs');  
app.locals.pretty = true;
```

//Ikooni kuvamine

```
app.use(favicon(path.join(__dirname, 'public/icons', 'favicon.ico')));
```

//Port seaded

```
app.set('port', process.env.PORT || 8080);  
app.use(logger('dev'));  
app.use(express.static(path.join(__dirname, 'public')));  
app.use(errorhandler());  
var scripts = [],
```

Lisa E järg

```
    styles = [],  
    sysscripts = [];  
app.get('/', function (req, res) {  
    var view = __dirname + '/views/full_screen_ui_theme/' + 'index.ejs';  
    res.render(view, {  
        title: 'KAUR juhtliides',  
        scripts: scripts,  
        styles: styles,  
    });  
});  
  
var camera = vapix.createCamera({  
    address: '192.168.1.12',  
    port: '80',  
    username: 'root',  
    password: 'Liveit11'  
});  
  
var options = {  
    resolution: '1920x1080',  
    compression: 60,  
    duration: 0,  
    fps: 0,  
    rotation: 180,  
    audio: 1,  
    clock: 1,  
    date: 1,
```

Lisa E järg

```
        textcolor: 'black',
        textbackgroundcolor: 'transparent',
        textpos: 1,
        text: 1
    }
var video = camera.createVideoStream(options);
video.on('data', function(data, error) {
    if (error) {
        console.log('Viga kaamera andmevoos: ');
        throw error;
    }
    else {
        var frame = new Buffer(data).toString('base64');
        io.sockets.emit('VideoStream', frame);
    }
});
video.on('end', function() {
    console.log('Kaamera andmevoog suletud');
});
serialport.list(function (error, ports) {
    if ( error ) {
        console.log ('Viga jadaliideste tuvastamisel');
    }
    else {
        console.log('Saadaval olevad COM portid:');
```

Lisa E järg

```
ports.forEach(function(port) {  
    console.log(port.comName);  
});  
}  
});  
  
//Avame jadasuhtluse porti  
var UARTPort = new SerialPort('COM9',  
    {  
        baudrate: 115200,  
        dataBits: 8,  
        parity: 'none',  
        stopBits: 1,  
        parser: serialport.parsers.readline("\r\n"),  
        flowControl: false  
    }, true);  
  
var KAURData = "";  
  
UARTPort.on('open', showPortOpen);  
  
UARTPort.on('data', handleReceivedSerialData);  
  
UARTPort.on('close', showPortClose);  
  
UARTPort.on('error', showSerialError);  
  
function showPortOpen() {  
    console.log('Kontroller USART ühendatud: ' + UARTPort.path);  
    console.log('Andmeside kiirus: ' + UARTPort.options.baudRate + ' baud');  
}  
  
var gpRegControlMode = false;
```

Lisa E järg

```
var gpTankControlMode = true;

io.on('connection', function (socket) {

    var port = 1500;

    var starboard = 1500;

    var vertical = 1500;

    var command;

    var rightStick_Y_State = 0;

    var rightStick_X_State = 0;

    var commandbuffer = new Buffer(11);

    var buttonbuffer = new Buffer(4);

    socket.on('button-down', function (data) {

        if (data.button == 'START') {

            command = 't';

            if (motoractive == false) {

                motoractive = true;

            }

            else if (motoractive == true) {

                motoractive = false;

            }

        }

        else if (data.button == 'BACK') {

            command = 'l';

            if (lightsactive == false) {

                lightsactive = true;

            }

        }

    }

}
```



```
else if (lightsactive == true) {  
    lightsactive = false;  
}  
}  
else if (data.button == 'X') {  
    command = 'x';  
    if (sensorsactive == false) {  
        sensorsactive = true;  
    }  
    else if (sensorsactive == true) {  
        sensorsactive = false;  
    }  
}  
else if (data.button == 'LB') {  
    command = '-';  
}  
else if (data.button == 'RB') {  
    command = '+';  
}  
else {  
    command = '?';  
}  
buttonbuffer.write('<',0);  
buttonbuffer.write('B',1);  
buttonbuffer.write(command,2);
```

Lisa E järg

```
buttonbuffer.write(';3);

console.log(buttonbuffer);

UARTPort.write(buttonbuffer, function(error, results) { //Andmete saatmine

    if ( error ) {

        console.log('Viga kontrolleri andmete saatmisel: ' + error);

    }

    else {

    }

});

});

socket.on('button-up', function (data) {

    console.log('Vabastatud nupp: ', data);

});

if (gpRegControlMode == true) {

    socket.on('axis-changed', function (data) {

        if (data.axis == 'RIGHT_STICK_Y' && data.value < 0) {

            rightStick_Y_State = data.value;

            port = 1500+(+(data.value * (-500)).toFixed(0));

            starboard = 1500+(+(data.value * (-500)).toFixed(0));

        }

        else if (data.axis == 'RIGHT_STICK_Y' && data.value > 0) {

            rightStick_Y_State = data.value;

            port = 1500-(+(data.value * 500).toFixed(0));

            starboard = 1500-(+(data.value * 500).toFixed(0));

        }

    }

}
```

Lisa E järg

```

    }

    else if (data.axis == 'RIGHT_STICK_X' && data.value > 0) {
        rightStick_X_State = data.value;
        port = 1500+(+(data.value * 500).toFixed(0));
        starboard = 1500-(+(data.value * 500).toFixed(0));
    }

    else if (data.axis == 'RIGHT_STICK_X' && data.value < 0) {
        rightStick_X_State = data.value;
        port = 1500-(+(data.value * (-500)).toFixed(0));
        starboard = 1500+(+(data.value * (-500)).toFixed(0));
    }

    else if (data.axis == 'LEFT_STICK_Y' && data.value < 0) {
        vertical = 1500+(+(data.value * (-500)).toFixed(0));
    }

    else if (data.axis == 'LEFT_STICK_Y' && data.value > 0) {
        vertical = 1500-(+(data.value * 500).toFixed(0));
    }

    else if (data.axis == 'RIGHT_STICK_X' || data.axis ==
'RIGHT_STICK_Y' && data.value == 0) {
        port = 1500;
        starboard = 1500;
    }

    else if (data.axis == 'LEFT_STICK_Y' || data.axis ==
'LEFT_STICK_X' && data.value == 0) {
        vertical = 1500;
    }

```

Lisa E järg

```

var checksum = port + starboard + vertical;

var command = '<' + 'M' + port + starboard + vertical + checksum;

console.log(command);

commandbuffer.write('<',0);

commandbuffer.write('M',1);

commandbuffer.writeInt16BE(port,2);

commandbuffer.writeInt16BE(starboard,4);

commandbuffer.writeInt16BE(vertical,6);

commandbuffer.writeInt16BE(checksum,8);

commandbuffer.write(';',10);

UARTPort.write(commandbuffer, function(error, results) { //Andmete
saatmine

    if ( error ) {

        console.log('Viga kontrolleri andmete saatmisel: ' +
error);

    }

    else {

    }

});

});

}

else if (gpTankControlMode == true) {

    socket.on('axis-changed', function (data) {

        if (data.axis == 'RIGHT_STICK_Y' && data.value < 0) {

Lisa E järg

            starboard = 1500+(+(data.value * (-500)).toFixed(0));

```

```

}

else if (data.axis == 'RIGHT_STICK_Y' && data.value > 0) {

    starboard = 1500-(+(data.value * 500).toFixed(0));

}

else if (data.axis == 'LEFT_STICK_Y' && data.value < 0) {

    port = 1500+(+(data.value * (-500)).toFixed(0));

}

else if (data.axis == 'LEFT_STICK_Y' && data.value > 0) {

    port = 1500-(+(data.value * 500).toFixed(0));

}

else if (data.axis == 'RIGHT_TRIGGER') {

    if (data.value < 0.05) {

        vertical = 1500;

    }

    else {

        vertical = 1500+(+(data.value * 500).toFixed(0));

    }

}

else if (data.axis == 'LEFT_TRIGGER') {

    if (data.value < 0.05) {

        vertical = 1500;

    }

    else {

        vertical = 1500-(+(data.value * 500).toFixed(0));

}

```

Lisa E järg

Lisa E järg

```

    }

    else if (data.axis == 'RIGHT_STICK_Y' || data.axis ==
'RIGHT_STICK_X' && data.value == 0) {

        starboard = 1500;

    }

    else if (data.axis == 'LEFT_STICK_Y' || data.axis ==
'LEFT_STICK_X' && data.value == 0) {

        port = 1500;

    }

    var checksum = port + starboard + vertical;

    var command = '<' + 'M' + port + starboard + vertical + checksum;

    console.log(command);

    commandbuffer.write('<',0);

    commandbuffer.write('M',1);

    commandbuffer.writeInt16LE(port,2);

    commandbuffer.writeInt16LE(starboard,4);

    commandbuffer.writeInt16LE(vertical,6);

    commandbuffer.writeInt16LE(checksum,8);

    commandbuffer.write(';',10);

    console.log(commandbuffer);

    UARTPort.write(commandbuffer, function(error, results) {

        if ( error ) {

            console.log('Viga kontrollile andmete saatmisel: ' +
error);

        }

        Lisa E järg

        else {

        }

    }

```

```
        });  
    });  
  
    }  
  
});  
  
//k2ivitame serveri  
server.listen(app.get('port'), function () {  
    console.log('Juhtliides kuulab KAUR porti: ' + app.get('port'));  
});
```

Lisa E järg

HTML

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
```

```
    <meta name="viewport" content="width=device-width">
```

```
    <title><%= title %></title>
```

```
    <link rel="stylesheet" href="/stylesheets/fonts.css">
```

```
    <link rel="stylesheet" href="/stylesheets/style.css">
```

```
    <link rel="stylesheet" href="/stylesheets/normalize.min.css">
```

```
    <% for(var i=0; i<styles.length; i++) { %>
```

```
      <link rel="stylesheet" href="<%= styles[i] %>"><% } %>
```

```
  </head>
```

```
  <script
```

```
src="https://ajax.googleapis.com/ajax/libs/jquery/1.8.2/jquery.min.js"></script>
```

```
  <script type="text/javascript" src="/socket.io/socket.io.js"></script>
```

```
  <script type="text/javascript" src="/javascripts/libs/gamepad.js"></script>
```

```
  <% for(var i=0; i<scripts.length; i++) { %>
```

```
    <script type="text/javascript" src="<%= scripts[i] %>"></script><% } %>
```

```
  <body>
```

```
    <div class="header-pane">
```

```
      <span class="name">KAUR</span>
```

```
    </div>
```

```
    <div class="top-pane">
```

```
  </div>
```



Lisa E järg

```
<div class="right-pane">

</div>

<div class="footer-pane">

</div>

<script>

    var app = new app();

</script>

<ul id="gamepads"></ul>

<!--Teade mida kuvatakse juhtpuldi puudumisel!-->

<p id="connect-notice">Jätkamiseks ühendage juhtpult!</p>

<script>

$(document).ready(function() {

    var gamepad = new Gamepad();

    var socket = io.connect();

    <!--Määrame mida teha, kui juhtpult on ühendatud!-->

    gamepad.bind(Gamepad.Event.CONNECTED, function(device) {

        console.log('Juhtpult ühendatud', device);

        <!--Kuvada vastav teade juhtpuldi puudumisest!-->

        if (gamepad.count() != 0) {

            $('#connect-notice').hide();

        }

        $('#gamepads').append('<li id="gamepad-' + device.index +

"><h1>Gamepad #' + device.index + ': &quot;' + device.id + '&quot;</h1></li>');

        var mainWrap = $('#gamepad-' + device.index),

            statesWrap,

            logWrap,
```

Lisa E järg

```
        control,
        value,
        i;

    mainWrap.append('<strong>State</strong><ul id="states-' +
device.index + "'></ul>');

    mainWrap.append('<strong>Events</strong><ul id="log-' +
device.index + "'></ul>');

    statesWrap = $('#states-' + device.index)
    logWrap = $('#log-' + device.index)

    for (control in device.state) {

        value = device.state[control];

        statesWrap.append('<li>' + control + ': <span id="state-'
+ device.index + '-' + control + "'>' + value + '</span></li>');

    }

});

<!--Määrame mida teha, kui juhtpulti pole ühendatud/-->
gamepad.bind(Gamepad.Event.DISCONNECTED, function(device) {

    console.log('Disconnected', device);

    $('#gamepad-' + device.index).remove();

    <!--Kuvame vastava teate, juhtpuldi puudumisel/-->

    if (gamepad.count() == 0) {

        $('#connect-notice').show();

    }

});

<!--Funktsioon nuppude staatuse jälgimiseks (pollimiseks)-->
gamepad.bind(Gamepad.Event.TICK, function(gamepads) {
```

Lisa E järg

```

<!--Deklareerime muutujad, kontrolleri nuppudel olekust/-->
var gamepad,

    wrap,

    control,

    value,

    i,

    j;

for (i = 0; i < gamepads.length; i++) {
    gamepad = gamepads[i];
    wrap = $('#gamepad-' + i);
    if (gamepad) {
        for (control in gamepad.state) {
            value = gamepad.state[control];

            $('#state-' + gamepad.index + '-' + control
+ ").html(value);

        }
    }
}

});

gamepad.bind(Gamepad.Event.BUTTON_DOWN, function(e) {
    socket.emit('button-down', { button: e.control });

    $('#log-' + e.gamepad.index).append('<li>' + e.control + '
down</li>');

});

gamepad.bind(Gamepad.Event.BUTTON_UP, function(e) {

    socket.emit('button-up', { button: e.control });

```

Lisa E järg

```

        $('#log-' + e.gamepad.index).append('<li>' + e.control + '
up</li>');

    });

    gamepad.bind(Gamepad.Event.AXIS_CHANGED, function(e) {

        socket.emit('axis-changed', { axis: e.axis , value: e.value });

        $('#log-' + e.gamepad.index).append('<li>' + e.axis + ' changed
to ' + e.value + '</li>');

    });

    if (!gamepad.init()) {

        alert('Teie veebilehitseja ei toeta juhtpulte, uuendage oma
veebilehitsejat.');
```

```
    }
```

```
});
```

```
</script>
```

```
</body>
```

```
</html>
```

## **LISA F - Käitursüsteemi testprogramm**

```

import serial

import serial.tools.list_ports as port_list

import time

import struct


data = []

startchar = '<M'

endchar = ';'


uart = serial.Serial(

    port='COM5',

    baudrate=115200,

    bytesize=serial.EIGHTBITS, #Sea bittide arv baidis

    parity=serial.PARITY_NONE, #Sea paarsus kontroll: no parity

    stopbits=serial.STOPBITS_ONE, #Sea stop bittide arv

    xonxoff=False, #Keela tarkvaraline flow control

    rtscts=False,

    dsrdtr=False

)

#Kuvame saada olevad COM portid

ports = list(port_list.comports())

print("Tuvastatud jadaliidese pordid:")

for p in ports:

    print (p)

if uart.isOpen():

    print("\n" + "Ühendus loodud pordiga: " + uart.port + "\n")

```

Lisa F järg

while True:

uart.flushInput() #Pühi uarti sisend puhver

uart.flushOutput() #Pühi uarti väljund puhver

#Küsida andmeid

$\mu$ s = input("Sisetage mootori signaali täitetegur (täisarv vahemikus 1000-2000 $\mu$ s): ")

if ( $\mu$ s.isdigit()) and (int( $\mu$ s) in range(1000, 2001)):

try:

#Saada andmed

$\mu$ s = int( $\mu$ s)

data.append( $\mu$ s & 0xFF);

data.append(( $\mu$ s >> 8) & 0xFF);

uart.write(startchar.encode())

uart.write(data)

uart.write(endchar.encode())

print("Saadetud käsk: <M" + str( $\mu$ s) + ";\n")

data.clear()

time.sleep(0.5) #Aega andmete vastuvõtuks

except Exception:

print ("Viga andmete saatmisel!")

else:

try:

if int( $\mu$ s) and ( $\mu$ s not in range (1000, 2001)): #Kui number vahemikust väljas

Lisa F järg

```
print("Sobimatus sisend! Täitetegur peab olema täisarv vahemikus 1000 kuni  
2000μs!\n")
```

```
except ValueError: #Kui pole number
```

```
if μs == 'x': #Kui "x", lõpeta programmi jooksumine
```

```
    break
```

```
else:
```

```
    print("Sobimatus sisend! Täitetegur peab olema täisarv vahemikus 1000 kuni  
2000μs!\n")
```

```
else:
```

```
    uart.close()
```

```
    print ("Viga jadaliidesega ühendamisel!")
```



**Lihtlitsents lõputöö salvestamiseks ja üldsusele kättesaadavaks tegemiseks ning juhendaja(te) kinnitus lõputöö kaitsmisele lubamise kohta**

Mina, \_\_\_\_\_,  
(*autori nimi*)

sünniaeg \_\_\_\_\_,

1. annan Eesti Maaülikoolile tasuta loa (lihtlitsentsi) enda loodud lõputöö

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_,  
(*lõputöö pealkiri*)

mille juhendaja(d) on \_\_\_\_\_,  
(*juhendaja(te) nimi*)

1.1. salvestamiseks säilitamise eesmärgil,

1.2. digiarhiivi DSpace lisamiseks ja

1.3. veebikeskkonnas üldsusele kättesaadavaks tegemiseks

kuni autoriõiguse kehtivuse tähtaja lõppemiseni;

2. olen teadlik, et punktis 1 nimetatud õigused jäävad alles ka autorile;

3. kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest tulenevaid õigusi.

Lõputöö autor \_\_\_\_\_  
(*allkiri*)

Tartu, \_\_\_\_\_  
(*kuupäev*)

---

**Juhendaja(te) kinnitus lõputöö kaitsmisele lubamise kohta**

Luban lõputöö kaitsmisele.

\_\_\_\_\_  
(*juhendaja nimi ja allkiri*)

\_\_\_\_\_  
(*kuupäev*)

\_\_\_\_\_  
(*juhendaja nimi ja allkiri*)

\_\_\_\_\_  
(*kuupäev*)